BASIC ATARI

EDIZIONE ITALIANA

Bill Carris



GRUPPO EDITORIALE JACKSON

BASIC ATARI

Bill Carris



GRUPPO
EDITORIALE
JACKSON
Via Rosellini, 12
20124 Milano

GRUPPO EDITORIALE
JACKSON

SUPERVISIONE TECNICA: Daria Gianni
GRAFICA E IMPAGINAZIONE: Francesca Di Fiore
COPERTINA: Marcello Longhini
FOTOCOMPOSIZIONE: CorpoNove s.n.c. - Bergamo
STAMPA: Grafika '78 - Via Trieste, 20 - Pioltello (MI)

Questo libro è dedicato a

NOME$*

per i vari livelli di aiuto che mi ha prestato senza porre condizioni.

* NOME$ è definito più avanti, a pagina 42

NOTA DEL TRADUTTORE

I programmi del testo sono stati provati su un ATARI 800XL collegato con l'unità a dischi ATARI 1050, utilizzando monitor e joystick commerciali: nel confronto con altri home computer, c'è da sottolineare l'assai raffinata gestione del suono e del colore.

Il modello 800XL presenta rispetto ai precedenti modelli 400 e 800 una tastiera con tasti meglio disposti e qualche scritta di poco differente.

Nel testo sono riprodotti fotograficamente i listati ottenuti con la stampante di qualità ATARI 1027; essi differiscono lievemente da quanto appare sul video, per il fatto che quest'ultimo dispone di 38 colonne, contro le 80 della stampante.

Inoltre, nell'introduzione dei programmi da tastiera, è possibile ricorrere ad abbreviazioni (terminate dal punto) che non compaiono più nella successiva visualizzazione o lista. Lasciando al lettore la scoperta – per tentativi – delle altre, citiamo:

POS. per POSITION
COL. per COLOR
SO. per SOUND
SE. per SETCOLOR
GR. per GRAPHICS
NE. per NEXT
PR. per PRINT (*)
IN. per INPUT

(*) oltre al classico "?"

SOMMARIO

INTRODUZIONE

Parte 1: Programmazione in BASIC

INTRODUZIONE

INTRODUZIONE

Lo scopo di questo volume è quello di introdurvi ai concetti fondamentali della programmazione in BASIC, nel modo più indolore possibile.

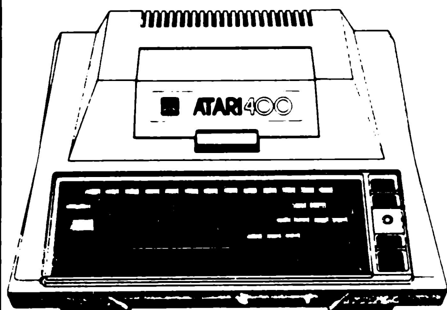
Questo è un testo operativo, da utilizzarsi insieme con un Home Computer ATARI: a lettura ultimata, dovreste possedere non solo una conoscenza iniziale del linguaggio BASIC, ma anche nozioni introduttive sulle capacità grafiche e sonore del Computer ATARI, nonché su altre interessanti caratteristiche di tale computer.

Ho volutamente evitato i *flow chart,* molti dettagli non indispensabili, numerose eccezioni alle regole, ed altri tradizionali approcci alla cultura informatica poiché, se da un lato sono un male inevitabile per il programmatore professionista, dall'altro tendono ad allontanare il principiante.

La programmazione professionale richiede un'attenzione precisa e noiosa per i dettagli, ma imparare a realizzare programmi elementari su un home computer è piuttosto un'esplorazione avventurosa e gradevole del legame uomo - macchina.

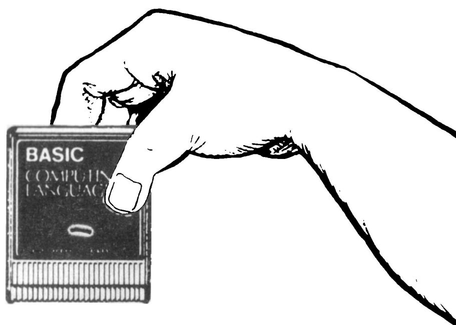
Niente di grave, se commettete un errore mentre imparate a far funzionare l'home computer: coloro che imparano dai propri errori e *"digeriscono"* questo libro ben presto morderanno il freno per informazioni e metodi più avanzati. Alcuni potranno perfino sviluppare la vocazione di diventare dei professionisti. Ma, prima di ogni altra cosa: benvenuti nel Basic ATARI.

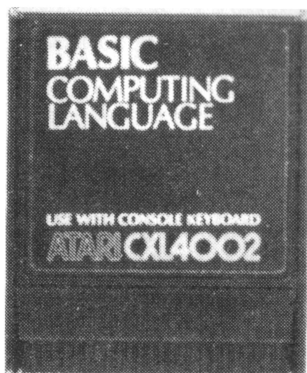
Questo libro è destinato per l'impiego sia con l'Home Computer ATARI 400 che con l'Home Computer ATARI 800,





insieme con la cartuccia del Basic ATARI.





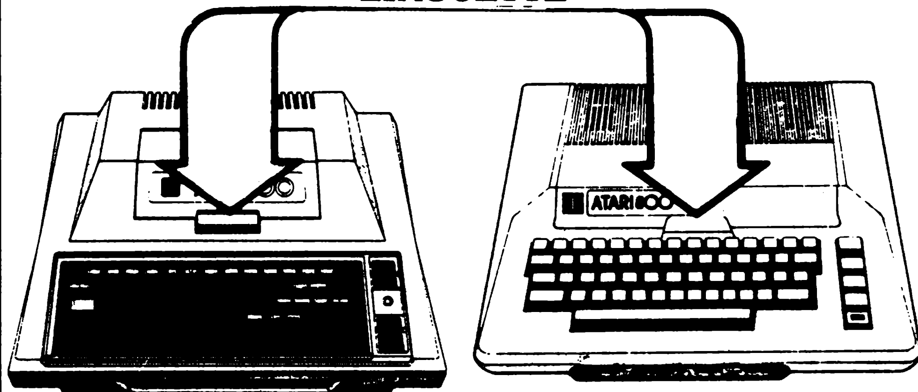
CHE COS'È IL BASIC?

Il Basic è un linguaggio per computer molto popolare, più facile da apprendere di molti altri.

Parte 1

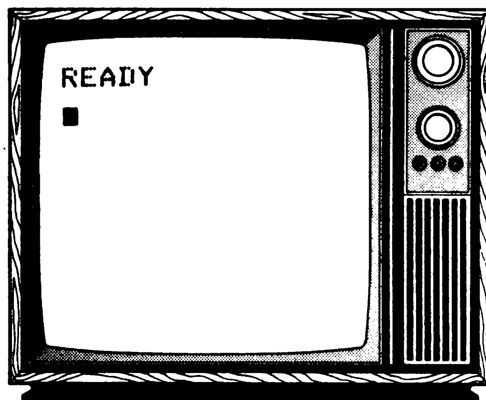
Programmazione in BASIC

Per inserire la cartuccia del Basic, per prima cosa aprite lo sportello del suo alloggiamento, tirando la linguetta in basso e verso di voi.

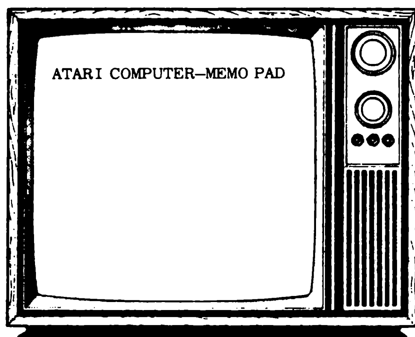


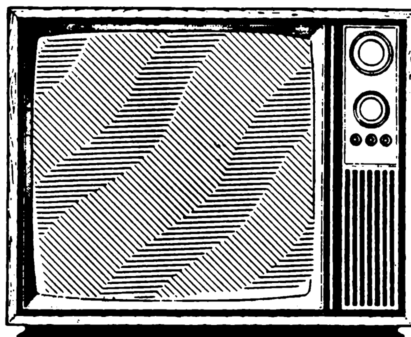
Quindi, con l'etichetta rivolta verso di voi, spingete la cartuccia lentamente ma con fermezza nella sua sede.

Quando chiudete lo sportello, accendete il computer ed il vostro televisore dovrebbe apparire così...



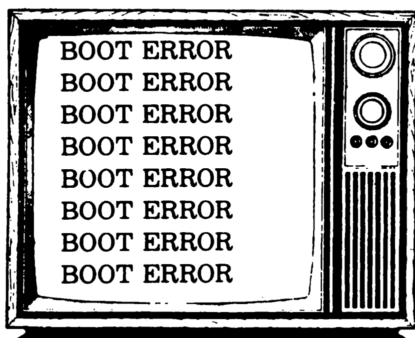
Se vi capita una delle situazioni seguenti...





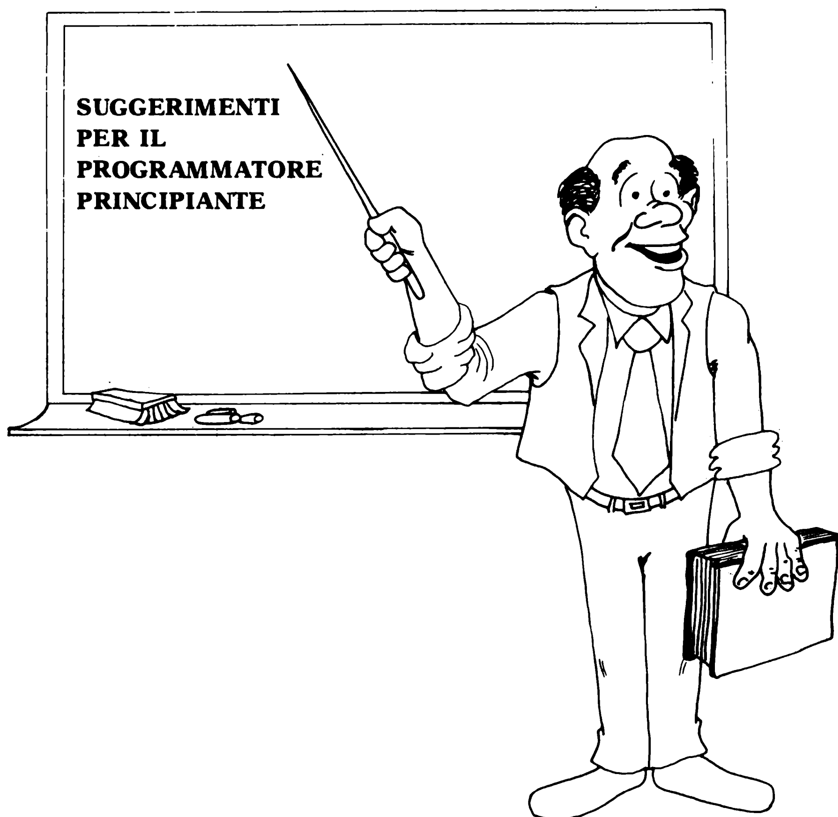
Potreste non aver premuto la cartuccia sufficientemente a fondo nella sua sede.

Probabilmente non avete gli interruttori 2-3 del computer ed il televisore posizionati sullo stesso canale. Provate a cambiare canale e controllate anche dietro il televisore la scatoletta dgli interruttori del computer, per accertarvi che sia commutata su "computer".



Probabilmente avete collegato un disk drive *(unità a disco)* e qualcosa di banale non va: per esempio, state utilizzando un dischetto sbagliato o avete dimenticato aperto lo sportello del drive. Se state da poco imparando il Basic, forse è meglio spegnere il disco. Spegnete e riaccendete il computer. Dopo esservi un po' introdotti al Basic potrete utilizzare il manuale del vostro disk drive per familiarizzarvi con le istruzioni principali del disco. Agli inizi lavorate senza disco.

NOTA: per ogni altro problema e per le istruzioni complete di avviamento del sistema, consultate la vostra guida dell'utente.

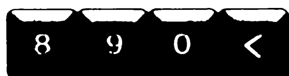


- Non temere! Nulla di quello che potrai fare con la tastiera danneggerà il computer o il televisore.
- Commettere degli ERRORI è una componente naturale dell'imparare a programmare. Tutti commettono errori e anche tu dovrai abituarti all'idea.
- Il tuo computer utilizza normalmente le LETTERE MAIUSCOLE; non accetterà istruzioni scritte in minuscolo.
- Inoltre, quando raggiungi il limite dello schermo continua imperterrito a scrivere: il computer va a capo automaticamente alla linea successiva.

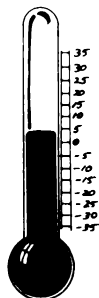
Poiché i computer sono creature molto speciali, è importante che non confondiate lo zero con la lettera O. Lo zero è in alto sulla tastiera e la lettera O gli cade molto vicino.

Perciò ricordate...

Questi sono ZERI



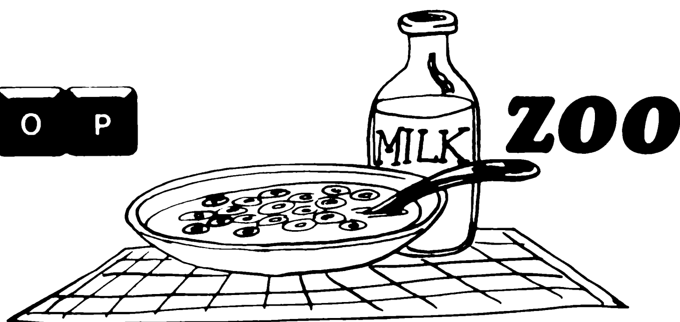






E queste sono LETTERE O

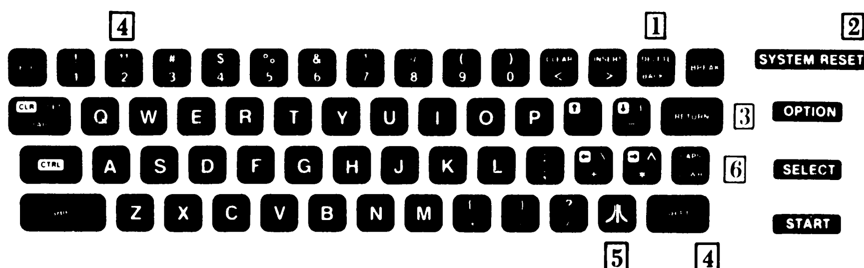




Se guardate attentamente lo zero sullo schermo TV, vedrete che ha nel centro un segno a forma di **S** che la lettera O non ha. Molte stampanti di computer e molta gente nei listati sbarrano lo zero (Ø) onde evitare confusione.

La tastiera

Sebbene la tastiera del vostro computer sia molto simile alla tastiera di una macchina per scrivere, ha alcune caratteristiche speciali: alcune le dovete imparare subito ed altre le apprenderete più tardi.



Questi sono i pochi tasti che dovete conoscere immediatamente. Provateli...

1 DELETE BACK S Questo tasto consente di cancellare ogni lettera battuta per caso. Digitate un po' di lettere e quindi ripercorretele a ritroso per afferrare il concetto.

2 SYSTEM RESET Questo tasto vi soccorre quando andate in confusione circa quello che sta succedendo sullo schermo. Finché non avrete familiarità con la tastiera, vi potrà capitare di creare per caso uno schermo pieno di messaggi d'errore o di caratteri incomprensibili. Premete SYSTEM RESET per cancellare uno schermo pasticciato e ripartire da zero.

3 Quando premete RETURN, voi dite al computer che state introducendo la vostra informazione o la vostra risposta.

Giusto per promemoria:

NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN

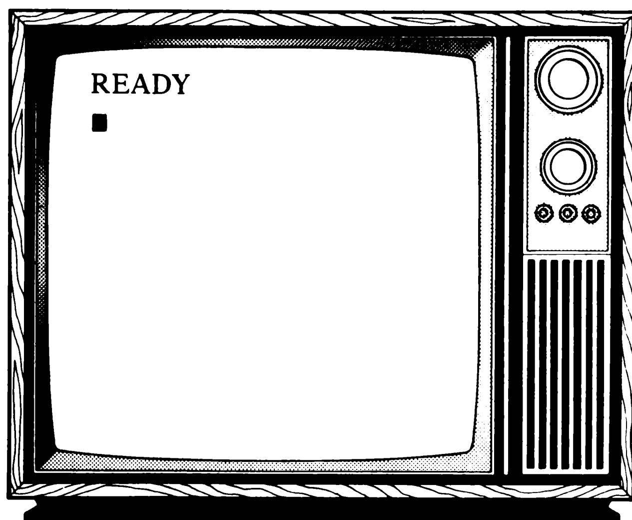
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN
NON SUCCEDE NULLA SE NON PREMETE IL TASTO RETURN

[4] Dovrete anche imparare alcune combinazioni di due tasti, che richiedono di mantenere premuto un tasto mentre se ne batte un altro. La prima combinazione è quella richiesta per stampare le virgolette. TENETE SCHIACCIATO il tasto SHIFT e contemporaneamente PREMETE " 2.

[5] Il tasto ⋀ vi mette in INVERSE VIDEO: premetelo e digitate qualcosa. Premetelo di nuovo e ritornerete alla rappresentazione normale.

[6] Il tasto CAPS LOWR vi mette a disposizione le lettere minuscole. Premetelo e battete un po' di lettere minuscole. Per tornare al maiuscolo dovete premere simultaneamente SHIFT e CAPS LOWR.

NOTA: l'inverse video e le lettere minuscole verranno utili più avanti; ma — come già detto — durante l'apprendimento lavorerete di norma con il MAIUSCOLO.



Quando il computer dice **READY** *(cioè "pronto")...*

ciò che esso è pronto a fare è prendere ordini da voi.

In effetti, scoprirete ben presto che tutto il trucco per scrivere i programmi in BASIC consiste nell'imparare gli ordini da dare al vostro computer.

Una delle istruzioni più comuni che utilizzerete si ottiene battendo le lettere...

PRINT

e dice al computer che volete scrivere qualcosa sullo schermo.

Quando digitate PRINT e premete il tasto RETURN, dite al computer di scrivere una linea bianca.

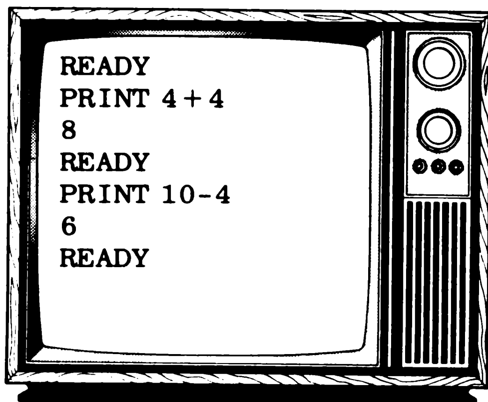
Esecuzione di calcoli semplici

PRINT 4+4 RETURN
PRINT 10–4 RETURN

NOTA: non è richiesto il segno di uguale.



Lo schermo dovrebbe apparire così.



Moltiplicare e dividere

* * * CRESCETE E MOLTIPLICATEVI * * *

Il simbolo della moltiplicazione col computer è l'asterisco *, ottenuto premendo il tasto [*].

Così non funziona....

```
PRINT 5 X 7
PRINT (5) (7)
PRINT 5 · 7
```

Così funziona...

```
PRINT 5 * 7 RETURN
```

e vi restituisce il risultato 35.

/ / / DIVIDE ET IMPERA / / /

Il simbolo della divisione col computer è la barra / ottenuta premendo il tasto [? /].

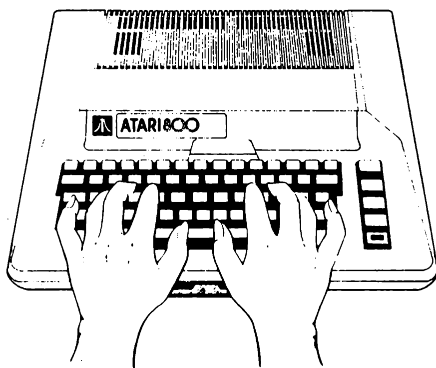
```
PRINT 99/3 RETURN
```

vi darà come risposta 33.

***********	QUESTO È IL SIMBOLO DELLA MOLTIPLICAZIONE
////////////	QUESTO È IL SIMBOLO DELLA DIVISIONE

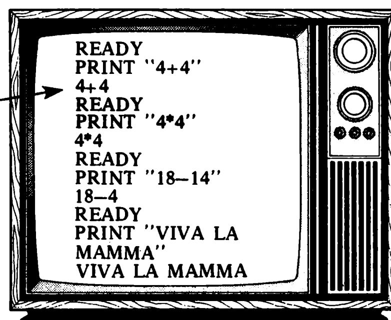
Scrivere lettere, parole e caratteri

PRINT "4+4" RETURN
PRINT "4*4" RETURN
PRINT "18-4" RETURN
PRINT "VIVA LA MAMMA" RETURN



Qualsiasi cosa tra le virgolette " " viene stampata direttamente.

Il risultato è stampato senza le virgolette. Tentare di produrre anche le virgolette sul video è possibile, ma farebbe venire un'ernia mentale al novizio.



NOTA: Potete abbreviare PRINT utilizzando il segno ?, ma i principianti probabilmente comprenderanno meglio i listati dei propri programmi scrivendo PRINT per esteso.

Modo diretto e modo programma

Con il vostro computer potete OPERARE sia SUBITO, sia PIÙ TARDI.

Modo Diretto

Istruzioni come PRINT "CIAO" RETURN sono eseguite non appena viene premuto il tasto RETURN e sono perciò dette di modo DIRETTO o IMMEDIATO.

Il modo DIRETTO è utile per avere risposte veloci e provare a vedere come il computer reagirà a certe istruzioni.

Modo Programma

Un programma è l'insieme di una o più istruzioni che non vengono eseguite finché non battete RUN RETURN.

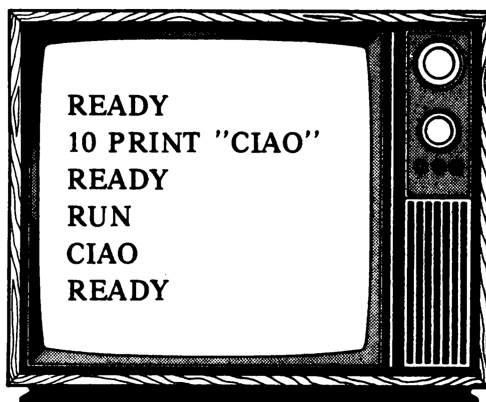
Ogni volta che scrivete RUN, il programma viene eseguito di nuovo.

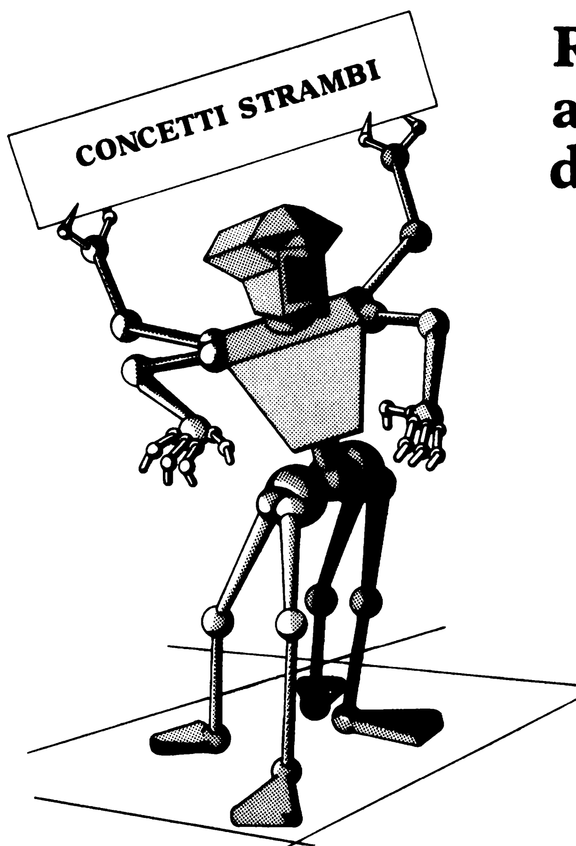
Questo è un Programma

10 PRINT "CIAO" RETURN
RUN RETURN

Osservate il numero 10 messo prima del comando di stampa.







Relativi ai numeri di linea

Ogni gruppo di istruzioni in un programma è preceduto da un numero di linea. Il calcolatore, se non è diversamente istruito, inizia ad eseguire il programma dalla linea di numero più basso e procede secondo i numeri crescenti.

```
10 PRINT "CIAO"
20
30
```

Molta gente, invece di numerare le linee 1, 2, 3 ecc., preferisce numerarle per esempio 10, 20, 30, ecc., onde lasciar spazio all'inserzione successiva di nuove linee.

Ancora sulle linee di programma ed i numeri di linea

Sostituzione di linee

Per il principiante, il modo più semplice di modificare una linea è riscriverla: la linea nuova sostituirà la vecchia.

10 PRINT "COME STA LA SIGNORIA VOSTRA" (vecchia linea)
10 PRINT "CIAO" RETURN (nuova linea)

A patto di usare lo stesso numero di linea, anche una linea più corta può sostituire quella originale.

Cancellazione completa di linee

È cosa da nulla. Scrivete il numero della linea indesiderata e premete RETURN

10 RETURN elimina la linea 10.

Il limite delle 3 linee

Nello scrivere programmi, vi viene richiesto un numero di linea nuovo almeno ogni 3 righe visualizzate sullo schermo.

```
10 PRINT "NON POTETE SCRIVERE PIÙ DI
3 LINEE DI INFORMAZIONE SUL VIDEO SENZ
A AGGIUNGERE UN NUOVO NUMERO DI LINEA"
20 PRINT "PER QUESTO MOTIVO È STATA A
GGIUNTA LA 20"
```

Fin qui vi è andata troppo bene...

Scoprirete che presto diventerà quasi una seconda natura il premere il tasto RETURN al termine dell'immissione dei dati.

D'ora innanzi, per rendervi più facili da leggere i listati e le istruzioni dei programmi, il tasto RETURN di norma non verrà più indicato.

QUESTA NON È TUTTAVIA UNA RITIRATA STRATEGICA. ECCO QUI ANCORA UN PO' DI RETURN, GIUSTO DA SOMMERGERVI.

RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN
RETURN RETURN RETURN RETURN RETURN RETURN

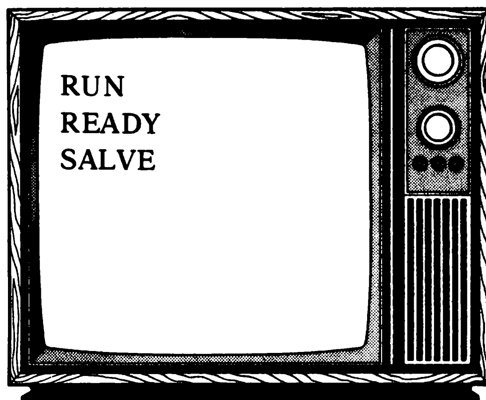
L'istruzione REMark

QUESTA ISTRUZIONE AIUTA VOI E L'ALTRA GENTE A CAPIRE I VOSTRI PROGRAMMI.

```
10 PRINT "SALVE"
20 REM QUESTO E' UN PROGRAMMA ASSOLUTAMENTE OTTUSO CHE MOSTRA IL COMPUTER CHE DI
CE SALVE
```



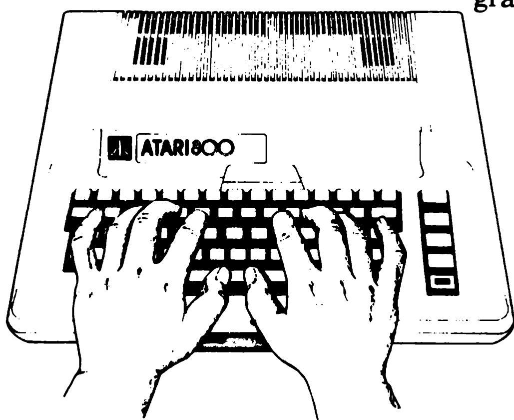
Se battete un numero di linea seguito da REMARK oppure REM, potete poi scrivere fino a tre righe di informazione che non influenzeranno il funzionamento del programma.



LIST ovvero L.

LIST

Scrivete semplicemente LIST per listare l'intero programma.





Lista di parti di programma

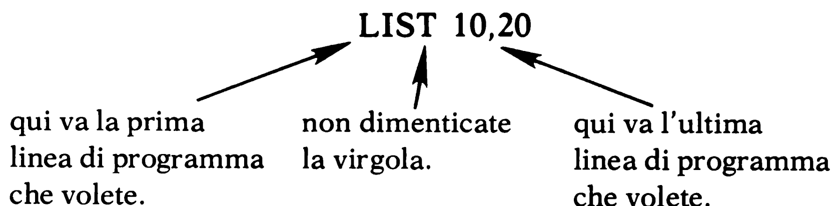
Quando comincerete a lavorare con programmi più lunghi, verrà il momento in cui vorrete listare una linea isolata o soltanto poche linee del programma.

Per listare una singola linea di programma, battete LIST e aggiungete il numero di linea prima di premere RETURN.

LIST 20 Sullo schermo apparirà solo la linea 20.

20 REM QUESTO È UN PROGRAMMA ASSOLUTAMENTE OTTUSO CHE MOSTRA IL COMPUTER CHE DICE SALVE

Per listare un certo intervallo di linee di programma, scrivete...



QUESTO PRODURRÀ IL LISTATO DELLE LINEE 10 E 20, NONCHÉ DI OGNI LINEA COMPRESA TRA QUESTE.

```
10 PRINT "SALVE"
20 REM QUESTO È UN PROGRAMMA ASSOLUTAMENTE
OTTUSO CHE MOSTRA IL COMPUTER CHE DICE SALVE
```

Più avanti, quando lavorerete con programmi più lunghi, apprezzerete la flessibilità di questi listati parziali.

Per listare un programma sulla stampante, battete...

LIST "P:

Stampa su carta con LPRINT

```
10 LPRINT "SALVE"
20 REM QUESTO E´ UN PROGRAMMA ASSOLUTAMENTE OTTUSO CHE MOSTRA IL COMPUTER CHE DI
CE SALVE
```



Per ottenere una copia durevole (su carta) utilizzate il comando LPRINT (LINE PRINT).

NOTA: Non potete usare LPRINT senza una stampante. Se tentate di farlo ugualmente otterrete il messaggio d'ERRORE n. 138, che significa che avete stampato senza una stampante.

Più semplice di così...



END



Alcuni tipi di programma richiedono che venga posta la frase END dove volete terminare l'esecuzione. Ciò non è sempre necessario sul Computer ATARI, ma è meglio esserne a conoscenza, così da poterla usare qualora vi servisse.



NEW

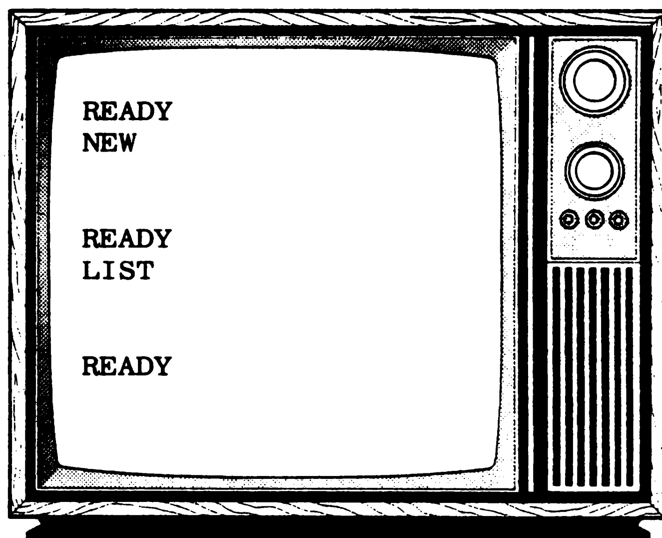
NEW



NEW cancella il vostro programma.

Senza possibilità di recupero.

Come potete vedere, anche scrivendo LIST, non compare alcun programma.



NOTA: Lo spegnimento del computer o l'apertura dello sportello della cartuccia produrranno anch'essi la cancellazione del programma. Invece, lo spegnimento del televisore non ha effetto.

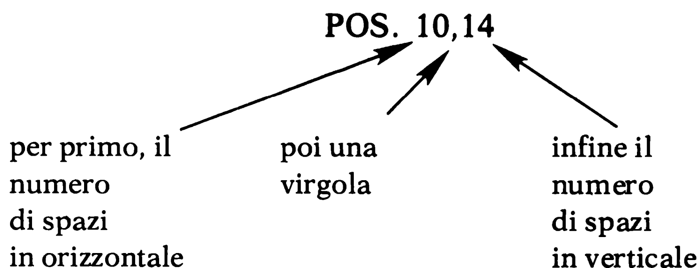
POSITION ovvero POS.

L'istruzione POSITION dice al computer dove partire nello scrivere sullo schermo.

Immaginate il vostro schermo suddiviso in 40 blocchi secondo la larghezza e in 24 secondo l'altezza.

Se prima di una PRINT non utilizzate la POSITION, il computer automaticamente inizia a scrivere sulla linea sottostante quella appena usata, spostato di due spazi rispetto al margine sinistro.

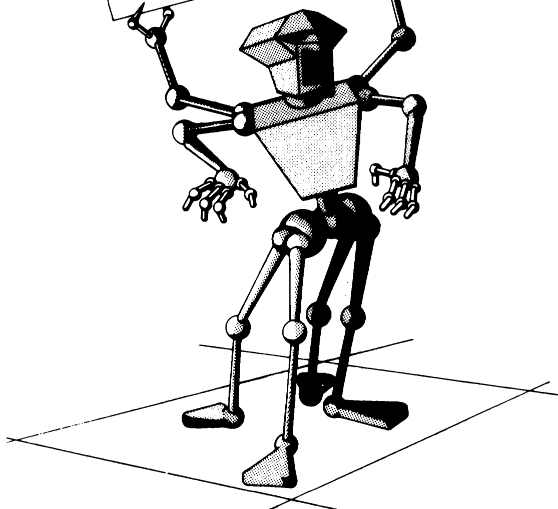
Con la POSITION, fornite sempre



In un programma, POSITION si usa come segue...

```
10 POSITION 0,0
20 PRINT "EHI"
30 POSITION 35,0
40 PRINT "LA´"
50 POSITION 16,11
60 PRINT "CENTRO DEL VIDEO"
```

Prima di mandare in esecuzione questo programma, premete SYSTEM RESET per cancellare lo schermo. In futuro apprenderete metodi più efficienti per eliminare la vista delle scritte indesiderate.

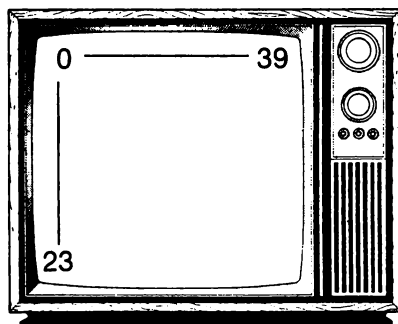


La numerazione del Computer

Per ragioni tecniche, i sistemi di numerazione interna del computer utilizzano come primo numero lo 0 invece dell'1.

Per esempio, se il computer ha 12 livelli grafici, essi saranno numerati 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, e 11.

Per l'uso della POSITION...



I 40 spazi orizzontali sono numerati da 0 a 39.
I 24 spazi verticali sono numerati da 0 a 23.

Operatori

Gli insegnanti di lettere sono tutti imbroglioni; vi hanno detto che questi...

: , ; sono segni di punteggiatura!

QUESTI SONO OPERATORI

I due punti :

Potete usare i due punti per infilare più di una istruzione in una linea di programma. Ciò vi consente di:

- risparmiare tempo
- risparmiare memoria
- organizzare meglio il vostro programma
- talvolta complicare orribilmente le cose.

Digitate NEW RETURN e quindi il seguente programma di una linea. Osservate che in una linea sono condensate tre istruzioni.

```
10 PRINT "STO PER SCRIVERE NEL BEL MEZZO DELLO SCHERMO":POSITION 6,11:PRINT "IL
MIO FATO SI E' COMPIUTO"
```

La seguente è una trappola diffusa:

```
10 REM QUESTA È LA PRIMA LINEA DEL PR
OGRAMMA DI MAGAZZINO: PRINT "ANALISI D
ELLE SCORTE E DEGLI ORDINI"
```

Nulla di quanto segue un'istruzione REM viene riconosciuto dal computer. Finché state imparando il BASIC, confinate le istruzioni REM in linee a sé stanti.

La virgola ,

La virgola vi aiuta ad organizzare l'incolonnamento delle vostre uscite. Se inserite delle virgole tra gli elementi da stampare, il computer mette le parole o i numeri in colonne distanziate di 10 posizioni.

```
10 PRINT "ATARI","PANNA","E","CIOCCOLATA"
20 PRINT "1","2","3","4"
```

ATARI	PANNA	E	CIOCCOLATA
1	2	3	4

NOTA: Nel capitolo sulle PEEK e POKE si può trovare l'informazione di come variare la spaziatura delle colonne.

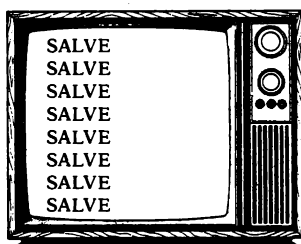
Il punto e virgola ;

Questo operatore vi consente di connettere più parti.

Provate il programma seguente, prima senza e poi con il punto e virgola.

```
10 PRINT "SALVE";
20 GO TO 10
```

NOTA: programmi BASIC come questo si arrestano premendo il tasto BREAK. Battendo CONT RETURN il programma riprende l'esecuzione.





L'istruzione 10 PRINT "PREZZO DELL'ORO=";"$385.98"
produrrà
PREZZO DELL'ORO=$385.98

Perché dobbiate fare tutto questo traffico diventerà evidente ben presto.

Messaggi d'errore I migliori amici che mai abbiate detestato

I Computer ATARI vi forniscono due tipi di MESSAGGI D'ERRORE assai utili.

Il primo tipo è visualizzato quando digitate una linea mancante della punteggiatura adatta o degli operatori leciti, o qualsiasi altra cosa priva di senso per il computer.

Ecco alcune "*papere*" comuni...

PRIMT "SALVE"

provoca

ERROR – PRIMT [“]SALVE"



a un esame accurato scoprite che PRINT è scritto in modo errato.

Quando vi capita questo tipo di messaggi d'errore, il computer piazza un contrassegno al termine della prima area dubbia (e non sull'errore vero e proprio); con un po' di pratica localizzerete che cosa non piace al computer.

PRINT SALVE"

provoca

ERROR – PRINT SALVE[”]

PRINT è scritto correttamente, ma mancano le virgolette prima della S di SALVE.

print "salve"

provoca

ERROR –[p]rint "salve"

Ricordate che non si possono usare le lettere minuscole per i comandi.

POS.2.5: PRINT "SALVE"
provoca

ERROR – POS.2.5[:]PRINT "SALVE"

Inesorabile: avete usato il punto (.) invece della virgola tra il 2 e il 5.

POS 2,5: PRINT "SALVE"
provoca

ERROR – POS [2],5: PRINT "SALVE"

ACCIDENTI! Avete messo la virgola al posto giusto, ma dimenticato il punto dopo POS.

10 "PRINT SALVE"
provoca

10 ERROR – ["]PRINT SALVE"

Lapsus freudiano: avete messo le virgolette prima della P invece della S.

Messaggi d'errore di programma

Vi buscherete un secondo tipo di MESSAGGI D'ERRORE quando – immaginando di aver introdotto il vostro programma, contenente però qualcosa di illogico – tenterete di mandarlo in esecuzione.

Il messaggio d'errore vi fornirà il numero corrispondente al tipo di errore nonché la linea ove il computer ha difficoltà nell'eseguire il vostro programma.

Quando ricevete un messaggio d'errore con un numero di lnea, come per esempio...

```
10 PRINT "SALVE"
RUN
ERROR 138 AT LINE 10
```
(errore 138 alla linea 10)

Per prima cosa guardate la linea specificata, per gli errori evidenti. Quindi ricercate il significato del tipo di errore: i più comuni sono elencati in fondo a questo libro. Se non lo trovate qui, consultate per la lista completa il Manuale di Riferimento del Basic ATARI.

Nel caso precedente ERROR 138 indica "*Device Time Out*": qualcuno ha utilizzato un comando LPRINT senza accendere la stampante.

Tra parentesi, il tutto può sembrare discretamente fastidioso, ma molti computer vi danno un bel pezzo meno d'aiuto nella ricerca degli inconvenienti.

Tre spunti matematici



Spunto n. 1

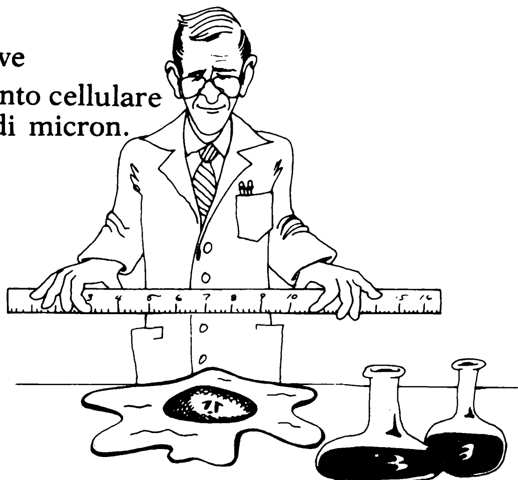
I computer utilizzano una notazione *condensata* per i numeri molto grandi o molto piccoli. Per il principiante, il modo migliore di averci a che fare è:

evitare l'uso di numeri **molto grandi**

O molto piccoli!

Ovviamente, non potete farlo se siete:

1. Un ricco sceicco petroliere, con alcuni miliardi da non perdere di vista.
2. Un microbiologo che deve misurare un accrescimento cellulare di uno 0.00000007-mo di micron.
3. Un masochista.



Ammettendo che siate il n. 3, ecco un breve riassunto di come tradurre la notazione condensata nell'usuale notazione umana.

I numeri MOLTO GRANDI condensati hanno:

1. Una "E"
2. Un segno "+"
3. Un numero dopo il segno più (+)

3.7E +11

Per convertire uno di tali numeri, come per esempio

3.7E + 11

1. Ricopiate il numero fino alla E

3.7

2. Partendo dal punto decimale (*n.d.t. corrispondente alla ns. virgola*), contate verso destra tante posizioni quante sono indicate dal numero dopo il segno più (11 nel nostro caso)

3.70000000000.

vecchia posizione del punto decimale

aggiungete il numero di zeri necessario

nuova posizione del punto decimale

370000000000.

Questo è il numero che il computer ha abbreviato come 3.7 E + 11.

I numeri MOLTO PICCOLI condensati hanno:

2.831E –04

1. Una "E"

2. Un segno "–"

3. Un numero dopo il segno meno (-)

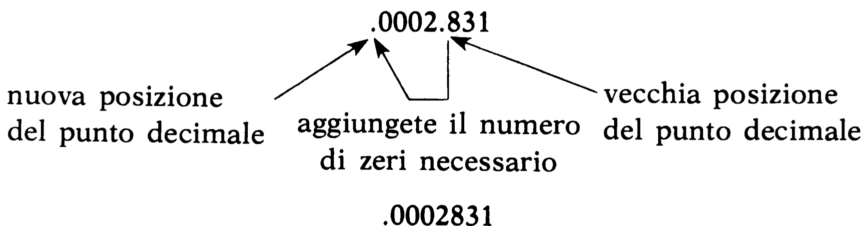
Per convertire uno di tali numeri, come per esempio

2.831E –04

1. Ricopiate il numero fino alla E

2.831

2. Partendo dal punto decimale, contate verso *sinistra* tante posizioni quante sono indicate dal numero dopo il segno meno (4 nel nostro caso)



.0002831

Questo è il numero che il computer ha abbreviato come 2.831E –04.

Spunto n. 2: sequenza di calcolo

Il computer calcola le espressioni matematiche in una certa sequenza. I calcoli dello stesso tipo o di uguale priorità sono eseguiti da sinistra a destra.

SIMBOLO

1. ( ) – per prima viene qualsiasi cosa tra parentesi; quindi

2. ∧ – l'elevamento a potenza, come per esempio: 5ˆ3 che è uguale a 5 per 5 per 5, ovvero 5^3 o 125; poi

3. */ – moltiplicazione e divisione, eseguite nell'ordine con cui si presentano da sinistra a destra; infine

4. + – – addizione e sottrazione, eseguite nell'ordine con cui si presentano da sinistra a destra.

Esempi:

Guardate quali differenze questo può comportare:

9 * 4 + 3 = 39

ma...

9 * (4 + 3) = 63.

Oppure...

$$62 - 3 * 8 = 38$$

ma...

$$(62 - 3) * 8 = 472$$

Se siete in dubbio, mettete le parentesi attorno a ciò che volete sia eseguito per primo.

Spunto n. 3

Ricordate quando, in quinta elementare, la maestra vi insegnava a scrivere i numeri con un puntino ogni tre posizioni decimali?

1.000.000 = un milione (secondo la maestra)

MAESTRA, TIENITI I TUOI PUNTINI!!!
IL COMPUTER NON LI USA!!!

1000000 = un milione (secondo il computer).

Ancora sulla tastiera



Magia del tasto CONTROL (CTRL)

Una volta appreso l'utilizzo del tasto CTRL, potrete realmente assumere il comando del video del vostro computer.

Il tasto CTRL va sempre tenuto abbassato mentre premete un altro tasto.

Per una gratificazione di un istante, premete simultaneamente i tasti CTRL e 2.

Questo capitolo è dedicato in particolare all'uso del tasto CTRL per correggere programmi e modificare ciò che appare sullo schermo.

Ricordate, quando vedete CTRL dovrete tenere abbassato il tasto CONTROL mentre premete l'altro tasto.

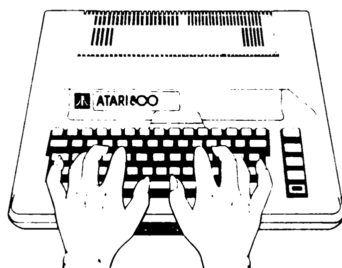
I passi che seguono servono per:

A. Muovere il cursore (segnalino del video) senza modificare il testo visualizzato.

B. Inserire e cancellare parole e spazi.

1. Iniziate con lo scrivere una linea di programma

10 PRINT "CIAO PIPPO"



2. Ora premete CTRL finché non avrete riportato il cursore sopra la cifra 1.

3. Ora premete CTRL e tenete schiacciato finché il cursore non è andato sull'iniziale di PIPPO. Se siete andati troppo oltre potete utilizzare CTRL per tornare indietro o insistere con CTRL lasciando che il cursore *"faccia il giro"* dello schermo ricomparendo a sinistra.

   Osservate che la vostra linea di programma rimane invariata; se non aveste tenuto abbassato il tasto CTRL essa sarebbe stata cancellata.

4. Rilasciate il tasto CTRL quando eseguite delle modifiche. Col cursore posizionato sulla P, levate il dito dal tasto CTRL e cambiate PIPPO in BIMBA. Premete RETURN e la modifica è compiuta.

5. Adesso, utilizzate come sopra i tasti CTRL e frecce, fermandovi questa volta tra CIAO e BIMBA.

   10 PRINT "CIAO BIMBA"
   fermatevi qui

6. Premete CTRL ed il tasto INSERT, che sta nella fila più alta della tastiera. Allontanate le parole inserendo degli spazi: anche fino

alla riga sotto, se volete. Potete tenere premuti CTRL e INSERT > per inserire un mucchio di spazi.

7. Premete una volta la barra spaziatrice e digitate la parola che volete inserire.

10 PRINT "CIAO BELLA BIMBA"

8. Premendo CTRL e il tasto DELETE BACK S la parola BIMBA verrà portata indietro vicino a BELLA.

9. Non dimenticate di premere RETURN, altrimenti le modifiche non verranno riportate nella memoria del computer.

L'impiego dei tasti CTRL e DELETE BACK S può essere molto avvincente.

Provate questa...

Utilizzate i tasti CTRL ↑ ← e frecce per portare il cursore, alla linea 10, sopra la B di BELLA.

10 PRINT "CIAO [B]ELLA BIMBA"

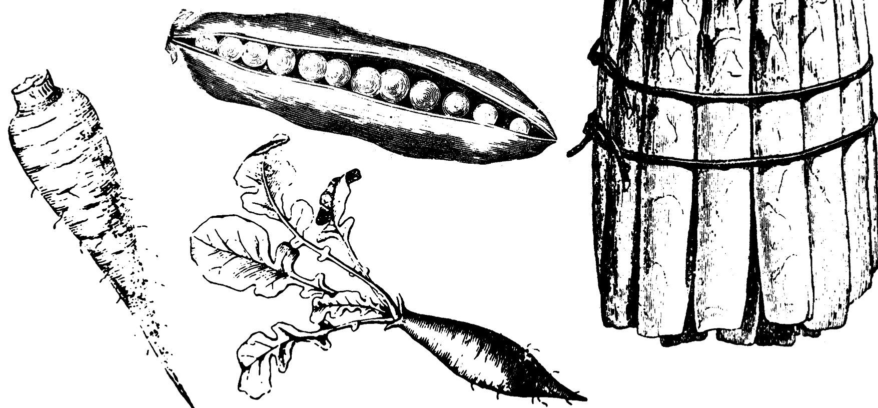
Ora premete CTRL DELETE BACK S ed osservate BELLA sparire sotto il cursore.

Se non premete RETURN dopo aver eseguito le vostre modifiche, risulta alterata solo la rappresentazione dello schermo e non la linea di programma nella memoria del computer. Se premete RETURN vi sbarazzate definitivamente del BELLA davanti a BIMBA.

AVVERTENZA FINALE. Probabilmente farete un bel po' di errori prima di acquisire la padronanza del tasto CTRL. Per esempio, vi capiterà di premere CTRL CLEAR invece di CTRL INSERT. Ma dopo un po' di pratica, apprezzerete veramente tutte queste possibilità di *"editing" (ingresso e correzione dati)* dello schermo.

Variabili

Questi sono vegetali, non variabili:



Queste sono variabili, non vegetali:

A TOT MESE DIMENS
NUM A$ NOMES$
RISP ETA QTA2 PIPPO BOH

Una variabile è qualcosa che può modificarsi o variare.

Esempi familiari di variabili sono:

Una nota musicale
Il colore di uno schermo TV
Il prezzo dell'oro
La vostra età
Il tasso d'inflazione

Utilizzando i computer, abitualmente si assegnano alle variabili dei nomi che aiutino a ricordarne il significato ed a lavorare con esse. Il prezzo dell'oro potrebbe essere chiamato PREZZORO o PREZZO o ORO o AURUM o OROPREZ o anche OTRANTO: al computer non importa proprio come lo chiamate.

SUGGERIMENTO: Non lasciate mai spazi nel nome della variabile: OROPREZ e non ORO PREZ.

Dopo che vi siete procurati il nome, vi serve un valore per la variabile.

LET OROPREZ = 395.63

Questa frase dice al computer di assegnare il valore 395.63 alla variabile di nome OROPREZ.

Adesso, domandategli PRINT OROPREZ e lui vi restituirà un valore di 395.63.



SI NOTI ANCHE: il computer parte assegnando il valore *zero* a tutte le variabili.

Provate così: PRINT BOH; scoprirete che BOH è zero fino a che non istruirete diversamente il computer.



Ci sono DUE tipi di variabili

Variabili numeriche

Nelle variabili numeriche mettete numeri:
PREZZO=395.63
ETA=22
VELOCITA=120

Variabili stringa

In queste mettete parole o caratteri:
NOME$="JOHN DOE"
MESE$="MAGGIO"
LOCAL$="EGITTO"
A$="DARIA"



Due cose importanti sulle Variabili Stringa

1. Per prima cosa DOVETE dire al computer quanto spazio riservare per la variabile stringa. Questo si ottiene con l'istruzione DIMENSION, che si presenta così:

```
10 DIM NOME$(8)
```

Ciò significa che avete fino ad otto caratteri per il nome.
"GIO" ci starebbe.
"CAROLINA" andrebbe bene.
"FRANZ SCHUBERT" sarebbe incompiuto.

2. L'altra cosa importante è ricordarsi di mettere le VIRGOLETTE ai lati delle vostre STRINGHE.

```
10 DIM A$(20) ←stiamo riservando più spazio
20 LET A$ = "FRANZ SCHUBERT"
```

non dimenticate di usare qui le virgolette

```
30 PRINT A$
```



Input

Un'istruzione che conferisce al computer una grande flessibilità è INPUT: si usa per farvi porre una domanda dal computer e consentire da parte vostra di fornire la risposta.

Di seguito è mostrato l'utilizzo di INPUT in un programma:

```
10 DIM CANE$(20)
20 PRINT "CHE SPECIE DI CANE POSSIEDI"
30 INPUT CANE$
40 PRINT "UN ";CANE$;" E´ UN BUON CANE"
```

Riguardo a INPUT, ricordate:

1. L'istruzione è normalmente utilizzata con un *"prompt" (messaggio di attenzione)* che guida l'utente a fornire il dato corretto.



Questo è il prompt:

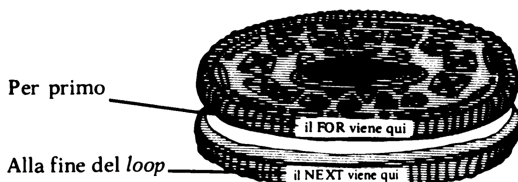
20 PRINT "CHE SPECIE DI CANE POSSIEDI"

2. Non occorre mettere un ? alla fine del prompt. L'istruzione INPUT lo fa per voi.

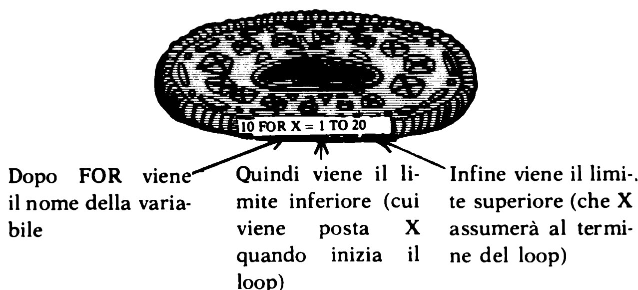
3. L'ingresso di una variabile stringa richiede che questa sia stata preventivamente DIMensionata.

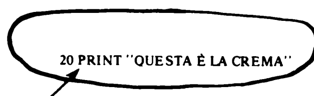
Cicli FOR/NEXT

Un ciclo *("loop")* FOR/NEXT è come un biscotto alla crema.

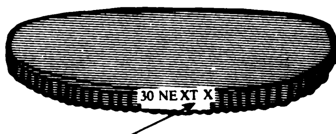


Tra di questi, c'è la crema del programma, ovvero ciò che volete ripetere utilizzando il ciclo. Vediamo il contenuto della porzione FOR:





Qui possono collocarsi una o più istruzioni lecite del Basic.



Questo deve essere lo stesso nome di variabile utilizzato nella parte FOR del *loop*.

Di nuovo il programma:

```
10 FOR X=1 TO 20
20 PRINT "QUESTA E´ LA CREMA"
30 NEXT X
```

Mandatelo in esecuzione. Quindi, aggiungete... **25 PRINT X**

Come potete vedere, il valore di **X** è incrementato di 1 ad ogni passo del loop.

Cambiate la linea 10 in... **10 FOR X = 1 TO 20000**
... e vediamo chi si stanca prima, se voi o il computer.



```
19899
QUESTA È LA CREMA
19900
QUESTA È LA CREMA
...
QUESTA È LA CREMA
19905
```

C'è un modo per far saltare al computer una certa quantità di numeri ad ogni passo del loop:

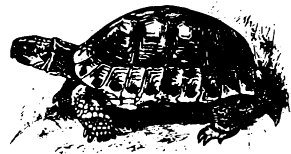
```
10 FOR Z=0 TO 100 STEP 5
20 PRINT Z
30 NEXT Z
```

questo dice al computer di contare di 5 in 5 da 0 a 100.

Potete anche andare all'indietro:

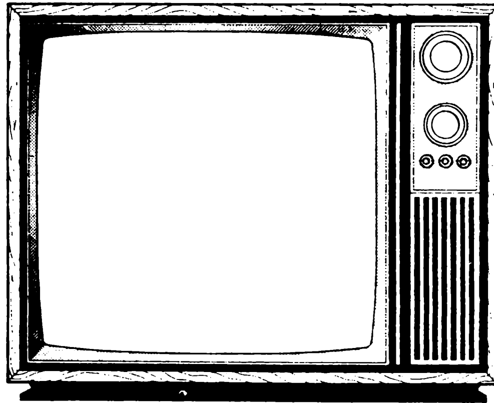
```
10 FOR Z=100 TO 0 STEP -5
20 PRINT Z
30 NEXT Z
```

non dimenticate il segno — (meno) andando dai valori più alti ai più bassi.



Rallentare il Computer con FOR/NEXT...

Talvolta il computer corre troppo veloce!!! Questo – per esempio – è il computer che dice CIAO *43 volte...*



VOLETE RIVEDERLO?

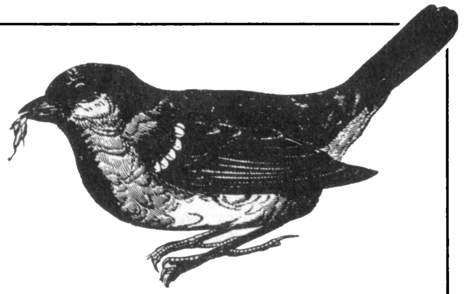
Per rallentare una visualizzazione, o qualsiasi altra cosa faccia il computer, potete utilizzare un loop FOR/NEXT con niente tra FOR e NEXT.

```
10 PRINT "BUON"
20 FOR RIT=1 TO 500
30 NEXT RIT
40 PRINT "GIORNO"
```

Quando usate un loop FOR/NEXT vuoto, obbligate il computer a compiere il ciclo senza far assolutamente nulla ad ogni passaggio:

"Computer: morditi la coda per 500 volte"

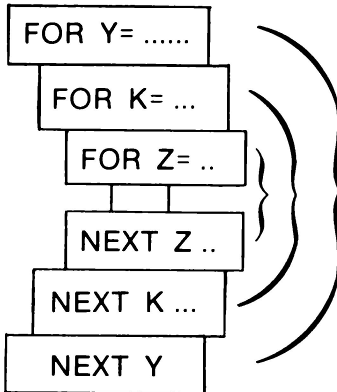
Provate a sperimentare valori diversi nella linea 20, osservandone gli effetti.



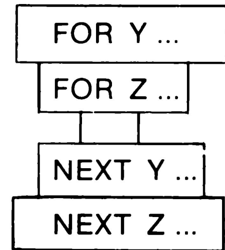
Loop FOR/NEXT annidati

"Annidare" un loop FOR/NEXT significa porlo dentro un altro. Potete far questo tutte le volte che vi serve, MA dovete disporre i FOR e i NEXT dall'interno verso l'esterno...

Questo funziona



Questo proprio no



Quello che segue è un esempio conclusivo:

```
10 FOR Y=1 TO 3
20 FOR K=255 TO 0 STEP -5
30 SOUND 0,K,10,10
40 PRINT K
50 FOR Z=1 TO 50:NEXT Z:REM RITARDO
60 NEXT K
70 NEXT Y
```

Se non riuscite a sentire niente, alzate il volume del televisore.

Ancora sulla Tastiera

Eccovi due ulteriori possibilità di *editing* dello schermo, con comandi da tastiera, che troverete utili:

I. Come arrestare temporaneamente i programmi Basic e come interrompere la lista di un lungo programma, per osservarne una parte. Per fermare programmi o listati tenete abbassato [CTRL] e premete [!/1]; con gli stessi tasti si riprende l'esecuzione. Provatelo con il programma seguente

10 PRINT "CIAO": GOTO 10

II. Come duplicare delle linee. Talvolta vi capiterà di ampliare il vostro programma al punto di trovarvi costretto a rinumerarlo. Più spesso, vi servirà ripetere una linea che già esiste o utilizzare una linea simile in un'altra parte del programma. Invece di ripetere la linea intera, riproducete quella che già avete e attuate le necessarie modifiche.

Ecco ciò che dovete fare per duplicare delle linee di programma...

```
30 FOR RIT=1 TO 50:NEXT RIT
```

Immaginate di avere questo meraviglioso loop di ritardo e di volerne uno uguale alle linee 70 e 90.

1. Tenete premuto [CTRL] ed usate [↑] per posizionare il cursore sul 3 di 30. Togliete il dito dal tasto [CTRL] e scrivete un 7 sopra il 3, quindi premete [RETURN]: non sembra sia successo nulla. Tornate un'altra volta in cima per cambiare il 7 in 9, e premete [RETURN].

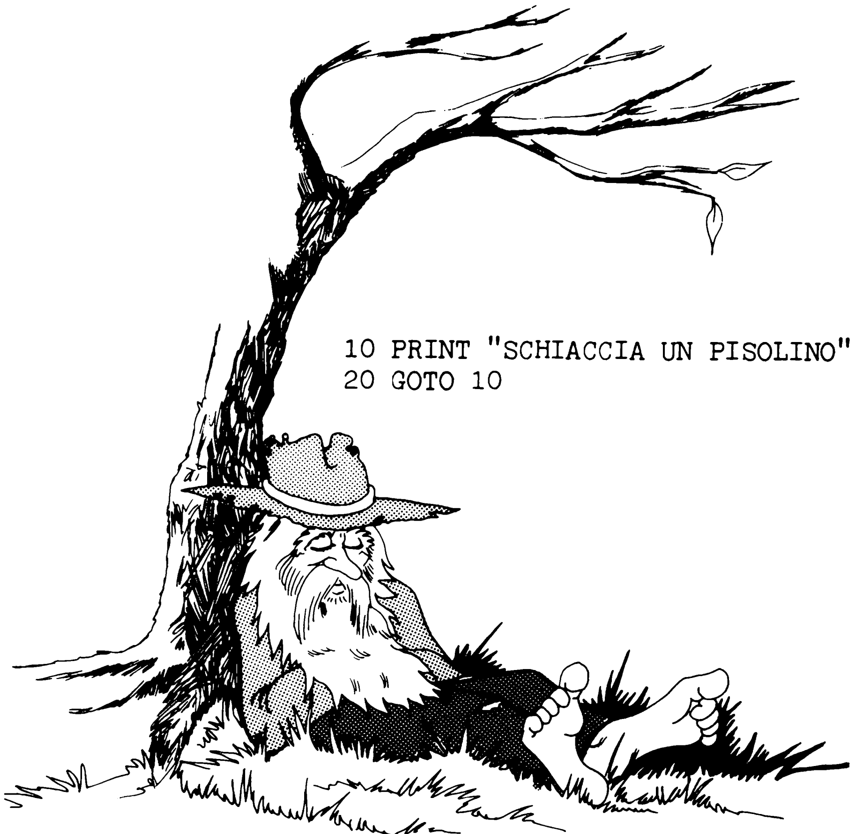
2. LISTate il programma per scoprire

```
30 FOR RIT=1 TO 50:NEXT RIT
70 FOR RIT=1 TO 50:NEXT RIT
90 FOR RIT=1 TO 50:NEXT RIT
```

Goto

L'istruzione GOTO è piuttosto semplice (e probabilmente una delle più grandi invenzioni che possiate incontrare nei computer).

```
10 PRINT "QUALCUNO TIRI LA CORDA"
20 GOTO 10
```



Incrementi, Contatori, Accumulatori

Gli INCREMENTI modificano il valore di una variabile ad ogni passo del loop. I CONTATORI registrano quante volte siete passati attraverso le istruzioni del ciclo.

Questo si chiama *inizializzare* una variabile (cioè assegnarle un valore iniziale) → 10 X=1

```
10 X=1
20 PRINT X
30 X=X+1

40 GOTO 20
```

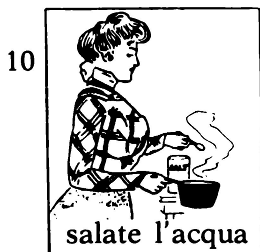
Questa non è un'equazione di tipo algebrico. In effetti significa: incrementa di 1 il valore attuale di X. → 30 X=X+1



Gli ACCUMULATORI memorizzano il valore corrente di un dato cumulativo.

```
10 REM PROGRAMMA ASSEGNI EMESSI
20 REM NUM E´ IL NUMERO ASSEGNO
30 REM IMP E´ L´IMPORTO
40 REM TOT E´ IL TOTALE PROGRESSIVO
50 NUM=1:IMP=0:TOT=0:REM INIZIALIZZAZIONE DELLE
VARIABILI
60 PRINT "IMPORTO DELL´ASSEGNO N. ";NUM;:INPUT IMP
70 NUM=NUM+1:TOT=TOT+IMP
80 PRINT :PRINT "IL TUO DEBITO E´ DI ";TOT;" LIRE"
90 PRINT :GOTO 60
```

Un loop di GOTO che non si ferma mai costituisce un *salto incondizionato*. Il seguente salto incondizionato è piuttosto ridotto ma produce risultati disastrosi.

10 

20 

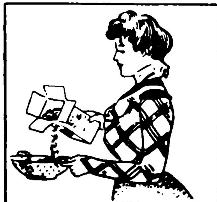


30 GOTO 10

Osservate ora che cosa è possibile fare combinando il GOTO con il salto condizionato prodotto da IF/THEN *(lett.: se/allora...)*

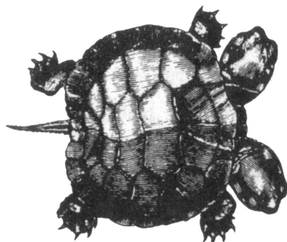
Att.: È un esempio,
non un programma vero.

10 SALARE L'ACQUA
20 ASSAGGIARE L'ACQUA
30 IF CONTENUTO-DI-SALE = 0,9 PARTI PER MILIONE
THEN PRINT "PERBACCO! È SQUISITO": GOTO 50
40 GOTO 10
50 CUOCERE ANCORA PER CINQUE MINUTI



60 PASSARE AL COLINO
ECC.

La capacità di decisione connessa a IF/THEN è la vera chiave di tutta la programmazione Basic; e come tutte le chiavi, dovete inserirla nel verso giusto.



Ci sono solo due modi di vedere: il confronto è vero o non lo è.

IF A>B THEN
significa (SE A È MAGGIORE DI B ALLORA...)

IF A<B THEN
significa (SE A È MINORE DI B ALLORA...)

IF A=B THEN
(EVIDENTE)

IF A<=B THEN
significa (SE A È MINORE O UGUALE A B ALLORA...)

IF A>=B THEN
significa (SE A È MAGGIORE O UGUALE A B ALLORA...)

IF A<>B THEN
significa (SE A NON È UGUALE A B ALLORA...)

IF A=X AND B=Y THEN
IF A=X OR B=Y THEN

Se il computer trova vero il confronto, esegue ciò che segue THEN.

10 IF A>B THEN PRINT "A È MAGGIORE"

Se il risultato del confronto è *"falso"*, il programma salta alla linea successiva.

20 PRINT "ALLORA A DEV'ESSERE MINORE O UGUALE A B"

```
10 PRINT :PRINT "DIMMI L´ETA´"
20 INPUT A
30 IF A>95 THEN PRINT "BENE, NON PRECISAMENTE UN GALLETTO DI PRIMO PELO":GOTO 10
40 PRINT "OH, BEATA GIOVENTU´!"
50 GOTO 10
```

Utilizzo di IF/THEN

Nel programma che segue, il computer controlla se voi immettete il numero 25. Se non lo fate, vi dà un'altra possibilità.

```
10 REM FUNZIONA SOLO IL 25.
20 PRINT "DIMMI UN NUMERO";:INPUT A
30 IF A<0 THEN PRINT "NUMERO NON VALIDO":GOTO 90
40 IF A<1000 THEN PRINT "FORSE CI SEI":PRINT :GOTO 60
50 PRINT "SEI PROPRIO LONTANO":GOTO 90
60 PRINT "ORA CONTROLLO":FOR RIT=1 TO 400:NEXT RIT
70 IF A<>25 THEN GOTO 90
80 PRINT :PRINT "CENTRATO!":END
90 PRINT :PRINT "PROVACI ANCORA":PRINT
100 GOTO 20
```



Non dimenticate l'esistenza dei confronti fra stringhe:

IF PAESE$="VIGGIU" THEN PRINT "FINALMENTE UN POSTO DECENTE PER VIVERCI!"

```
10 DIM NOME$(30)
20 PRINT "IL TUO NOME DI BATTESIMO ";
30 INPUT NOME$
40 IF NOME$="CESIRA" THEN GOTO 60
50 GOTO 20
60 PRINT :PRINT "BENVENUTA NEL SISTEMA, ";NOME$
```



e possiamo combinare confronti fra **STRINGHE E NUMERI**

```
10 DIM NOME$(30)
20 PRINT "IL TUO NOME DI BATTESIMO ";
30 INPUT NOME$
40 IF NOME$="CESIRA" THEN GOTO 60
50 GOTO 20
60 PRINT :PRINT "BENVENUTA NEL SISTEMA, ";NOME$
70 PRINT "QUAL E' IL TUO NUMERO DI IDENTIFICAZIONE";
80 INPUT IDEN
90 IF IDEN=472 THEN GOTO 110
100 PRINT "NUMERO ILLEGALE, RIPROVA!":GOTO 80
110 PRINT :PRINT "ACCOLTO"
120 PRINT "SCEGLI UN'OPZIONE"
```

ON/GOTO

L'istruzione ON/GOTO vi permette di eseguire facilmente più confronti insieme; il suo formato è il seguente:

ON X GOTO ______ , ______ , ______ ECC.

- X: una qualsiasi variabile
- primo campo: linea da eseguire se X=1
- secondo campo: linea da eseguire se X=2
- terzo campo: se X=3

Esempio:

È assai utile per ingressi di tipo multi-scelta.

```
10 PRINT "1=LINEA 100"
20 PRINT "2=LINEA 150"
30 PRINT "3=LINEA 200"
40 PRINT "4=LINEA 250"
50 PRINT :PRINT "BATTI UN NUMERO E PREMI IL TASTO
RETURN"
60 INPUT X:IF X>4 THEN GOTO 10
70 ON X GOTO 100,150,200,250
100 PRINT "QUESTA E´ LA LINEA 100":GOTO 300
150 PRINT "QUESTA E´ LA LINEA 150":GOTO 300
200 PRINT "QUESTA E´ LA LINEA 200":GOTO 300
250 PRINT "QUESTA E´ LA LINEA 250"
300 PRINT :FOR RIT=1 TO 200:NEXT RIT:RUN
```



NOTA: Quando il computer incontra l'istruzione RUN in un programma, riprende l'esecuzione dall'inizio.

GOSUB e RETURN

GOSUB e RETURN sono parenti stretti dell'istruzione GOTO, ma vi è un'importante differenza rispetto a questa:

> GOSUB tiene nota del punto da cui è saltata e, quando trova RETURN, ritorna all'elemento immediatamente successivo nel programma e lo esegue.

Per esempio:

```
10 PRINT "CIAO"
20 GOSUB 100
30 PRINT "ARRIVEDERCI"
40 END
100 FOR Z=1 TO 10:PRINT "FELICE DI VEDERTI":NEXT
Z:RETURN
```

Ora, aggiungete un'altra linea:

```
5 PRINT "BUONGIORNO":GOSUB 100
```

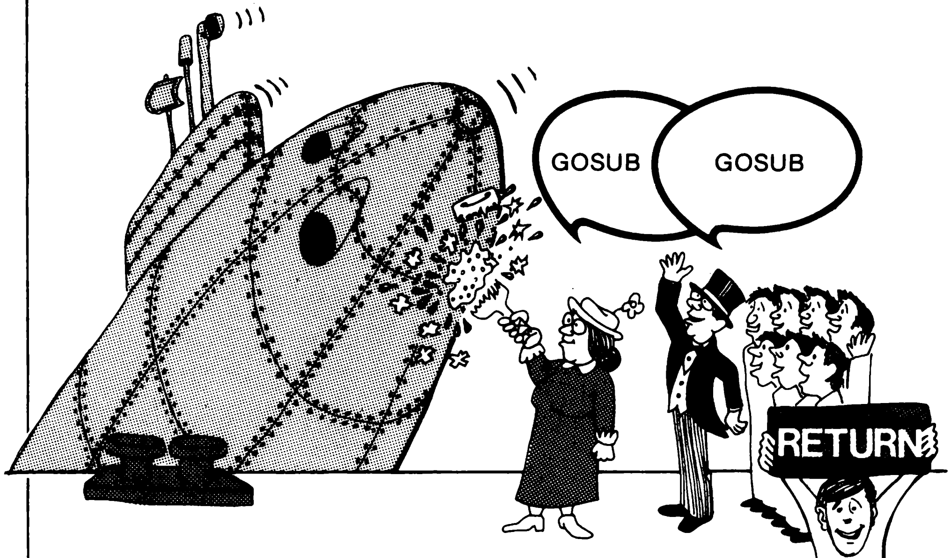


NOTA: L'END di linea 40 è necessario per impedire che il programma prosegua rovinosamente sulla linea 100.

Nel seguente programma, che calcola una media di numeri, la "SUBROUTINE" *(sottoprogramma)* aggiunge al totale ogni numero nuovo.

```
10 PRINT "QUANTI NUMERI DA MEDIARE ";
20 INPUT N
30 FOR Z=1 TO N
40 PRINT "NUMERO ";Z;
50 INPUT NUM
60 GOSUB 200
70 NEXT Z
80 MED=TOT/N
90 PRINT "LA MEDIA E´ ";MED
100 END
200 TOT=TOT+NUM:RETURN
```

Ovviamente, stavolta si poteva fare a meno della SUBROUTINE; ma ricordatevi che tra la linea 200 e il RETURN potreste mettere un numero elevatissimo di conti.



ON/GOSUB

ON/GOSUB è molto simile a ON/GOTO; tuttavia questa istruzione di salto contiene un RETURN che ON/GOTO non contiene.

Quello che segue è un esempio di ON/GOSUB.

```
10 REM ONGOSUB
20 PRINT "DOMANDA 1"
30 PRINT "QUAL E´ LA RADICE QUADRATA DI 25"
40 INPUT RQ:IF RQ=5 THEN BRAVO=INT(RND(0)*3+1):ON BRAVO GOSUB 1000,2000,3000,400
0:GOTO 70
50 PRINT :PRINT "RIPROVACI":GOTO 30
60 REM SOLO UN´ALTRA DOMANDA
70 PRINT :PRINT "DOMANDA 2"
80 PRINT "QUANTO DISTA IN MILIONI DI KM IL SOLE DALLA TERRA"
90 INPUT DS:IF DS=150 THEN BRAVO=INT(RND(0)*3+1):ON BRAVO GOSUB 1000,2000,3000,4
000:GOTO 110
100 PRINT :PRINT "RIPROVACI":GOTO 80
110 PRINT :PRINT "ECC.":END
1000 PRINT "GRANDIOSO!":RETURN
2000 PRINT "BEN FATTO!":RETURN
3000 PRINT "NOVELLO EINSTEIN!":RETURN
4000 PRINT "ANCORA GIUSTO!":RETURN
```

In questo esempio, se la persona dà la risposta giusta, il computer sceglie a caso uno dei 4 complimenti.

NOTA: per maggiori informazioni sull'utilizzo dei numeri casuali consultate il capitolo sulle Funzioni.

READ/DATA

```
10 READ X
20 DATA 20,7,5 100,40

40 GOTO 10
```

I computer non leggono per divertimento; occorre insegnargli a leggere. Fortunatamente, per far ciò non ci sono grossi problemi; dovete semplicemente dire al computer in modo estremamente diretto...

READ X (qualsiasi nome di variabile numerica o stringa) *read = leggi*

Il computer troverà il valore di X in un'istruzione DATA.

DATA 5,10,15,20,25

Per separare i valori nelle frasi DATA si utilizzano delle virgole. Eccone un esempio:

```
10 READ X
20 PRINT X:IF X=25 THEN END
30 GOTO 10
40 DATA 5,10,18,200,25
```

Qui non si usano le virgole.

Il *POINTER (puntatore al contenuto del DATA)* è alquanto pignolo e non perde il segno di dove è arrivato a leggere.



Il pointer inizia nella prima posizione che segue DATA.

40 DATA 5, 10, 18, 200, 25



Al secondo passo del loop, viene puntato il valore successivo.

40 DATA 5, 10, 18, 200, 25

Se non avete spazio sufficiente per tutti i valori da mettere in un DATA, potete creare una nuova linea DATA.

```
10 READ X
20 PRINT X
30 GOTO 10
40 DATA 5,10,18,200,25
50 DATA 34,44,5,56,2,344,333,5,5,666,2,2,37,86,5,84,35,66,6,88,57,7777,33,3,333,
9,4,7,34,85,3,44,46,3,6,3,6,9
60 DATA 2,34,55,66,77,-1
```

Pensate dover assegnare tutti questi valori con l'istruzione LET!

Come già detto, il **POINTER** è un vero rompiscatole, e se voi debordate dal DATA, vi mostra tutto il suo disappunto...

Quando il **POINTER** si trova a puntare il vuoto, anzichè un valore per la variabile, vi arriva

> "ERROR–6 AT LINE 10" *(errore –6 alla linea 10)* o qualche stramberia simile.

La maniera tradizionale per cavarsela è utilizzare un elemento di DATA con funzione di *FLAG (lett.: bandiera)* insieme con un'istruzione IF-THEN.



Un FLAG in un'istruzione DATA è un valore assai poco probabile a verificarsi, a meno che non vi sia posto intenzionalmente. È messo al termine dell'istruzione DATA per informare il computer che ha raggiunto la fine di tale istruzione.

Questo è un esempio di flag in un programma con frase DATA.

```
10 REM KPL=CHILOMETRI PER LITRO
20 READ KPL
30 IF KPL=100 THEN END ←—QUI IL CONFRONTO SUL FLAG
40 PRINT KPL;" KM/L"
50 DATA 5,8,10,12,14.5,20,100← VALORE IMPROBABILE
60 GOTO 20
```

Restore

Prima di abbandonare le istruzioni READ e DATA, occorre affrontare ancora qualche argomento: RESTORE, per esempio, che porta indietro il POINTER fino all'inizio.

Non solo è spesso estremamente utile, ma è indispensabile in programmi quale quello d'inventario che vi presentiamo più avanti.

Nel seguente programma sonoro, la linea 20 comanda al computer di eseguire una RESTORE e tornare all'inizio dei dati, dopo aver incontrato il FLAG—888.

```
10 READ X
20 IF X=-888 THEN RESTORE :GOTO 10
30 SOUND 0,X,10,10:PRINT X
40 FOR RIT=1 TO 40:NEXT RIT
50 GOTO 10
60 DATA 23,55,76,33,6,4,200,165,20,3,45,55,76,23,78,-888
```

Ricordatevi di alzare il volume del televisore.

CALMA, C'È DELL'ALTRO...

Nell'istruzione RESTORE potete aggiungere la specifica di un numero di linea ...

RESTORE 3000

Questa frase spiega al computer di posizionare il pointer sul primo valore dell'istruzione DATA che comincia alla linea 3000. Ovviamente, ce ne saranno anche di più!

Il computer può inseguire variabili diverse.

```
10 READ X,Y
20 PRINT X,Y
30 GOTO 10
40 DATA 5,10,8,9,7,6
```

Questi valori sono per la Y

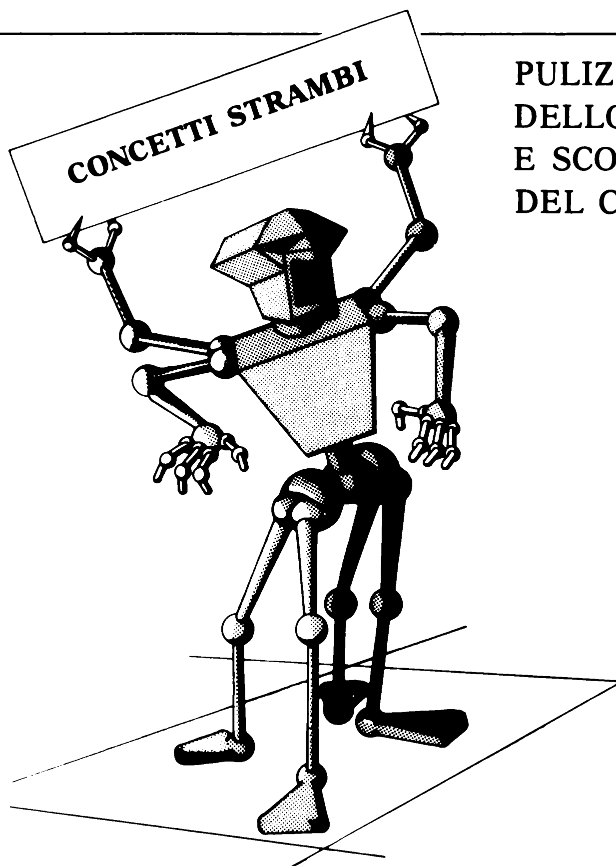
Questi valori sono per la X

E C'È ANCORA DELL'ALTRO! Potete usare le strighe nelle istruzioni DATA.

```
10 DIM NOME$(20)
20 READ NOME$
30 IF NOME$="GORGONZOLA" THEN END
40 PRINT NOME$
50 GOTO 20
60 DATA DARIA,DAVIDE,TOM,MARIA,GORGONZOLA
```

Infine, potete mescolare variabili stringa e variabili numeriche nella stessa frase READ. Il seguente è un semplice programma d'inventario.

```
10 REM PN=NUMERO DI PARTICOLARE, D$=DESCRIZIONE
20 DIM PARTIC$(20),D$(20)
30 PRINT "NOME DEL PARTICOLARE ":INPUT PARTIC$
40 RESTORE
50 READ D$,PN
60 IF D$="IPPOCAMPO" THEN GOTO 90
70 IF D$=PARTIC$ THEN PRINT "IL NUMERO DI PARTICOLARE E´ ";PN:PRINT :GOTO 30
80 GOTO 50
90 PRINT "NON C´E´":PRINT :GOTO 30
100 DATA RUOTA,1234,BULLONE,2345,PULEGGIA,3456,CATENACCIO,5678,NOTTOLINO,6789,IP
POCAMPO,7890
```



PULIZIA INIZIALE DELLO SCHERMO E SCOMPARSA DEL CURSORE

Molte volte desiderate immettere parole e caratteri sullo schermo, senza che sul video rimangano il READY e il cursore.

Per cancellare lo schermo in apertura di programma, iniziate con una sequenza piuttosto strana di tasti...

```
5 PRINT "↖"
```

Per ottenere questa freccia, premete una volta [ESC], quindi premete [CLEAR <] con abbassato il tasto [CTRL]; infine chiudete le virgolette.

Questa istruzione dice al computer di iniziare il programma con la cancellazione dello schermo.

Provate...

```
5 PRINT "↖"
10 GOTO 5
```

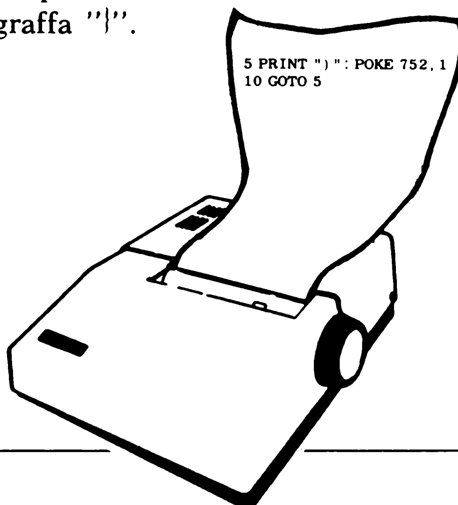


Osservate che lo schermo è ripulito, ma il cursore continua a comparire. Lo possiamo facilmente eliminare modificando come segue la linea 5

```
5 PRINT "↖": POKE 752,1
```

Con questa linea in cima al vostro programma potete rendere le cose assai più pulite.

NOTA: Quando listate su stampante un programma contenente un PRINT "↖", al posto della freccia vedrete su carta una parentesi tonda")" o graffa "}".



"Hip Hip Arrays"

5
10
15
20
30

Gli *ARRAY (lett.: schiere; liberam.: tabelle, vettori, matrici)* possono essere concepiti come contenitori

ovvero "oggetti" in cui si immagazzinano dei numeri. Nel Basic ATARI, gli array sono gruppi di variabili numeriche; come per ogni variabile numerica, potete assegnar loro dei valori, cambiarli, eseguire con essi manipolazioni aritmetiche (per esempio, moltiplicando il valore di un elemento per quello di un altro). Nell'esempio di figura, potreste moltiplicare la sommità del contenitore con il basso, ottenendo 150.

Ovviamente, come ogni cosa legata alla programmazione dei computer, occorre imparare alcune regole...

Regola n. 1

Ciascun elemento di un array ha un nome composto di due parti: vengono dapprima una o più lettere,

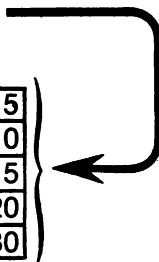
A(1)

quindi un numero (detto anche *indice*).

Ammettendo ora di aver specificamente assegnato all'array i valori di cui sopra, risulta:

A(1) uguale a 5
A(2) uguale a 10
A(3) uguale a 15
ecc.

A(1)
A(2)
A(3)
A(4)
A(5)



Regola n. 2

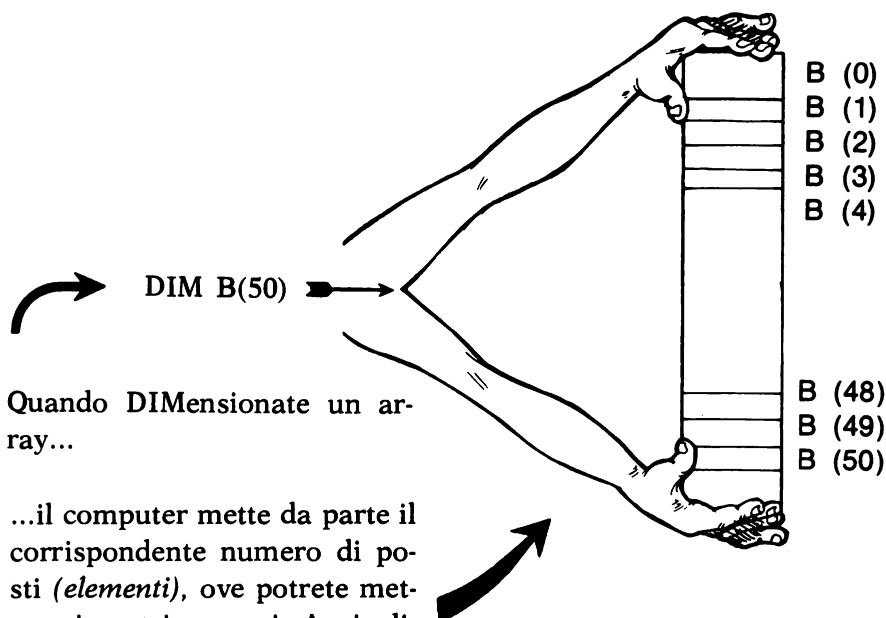
Prima di operare con gli ARRAY, occorre DIMensionarli. C'è tuttavia una differenza importante tra il dimensionamento degli array e quello delle variabili stringa.

Quando dimensionate un array, per esempio...

DIM B(50)

non state dicendo il numero di caratteri che debbono entrare nell'array (come sarebbe dimensionando una stringa); bensì informate il computer sul numero dei posti in cui andrete a mettere i numeri e sul nome ad essi assegnato.

Ecco un esempio di impiego...



Quando DIMensionate un array...

...il computer mette da parte il corrispondente numero di posti *(elementi)*, ove potrete mettere i vostri numeri. Anzi, disporrete di un elemento in più, poiché può essere usato anche B(0).



In primo luogo, con gli ARRAY è rapidissimo assegnare valori alle variabili; infatti, è più facile attribuire valori a un lotto di elementi di un array piuttosto che a un insieme di variabili separate.

```
10 A1=1
20 A2=2
30 A3=3
ecc.
```

Pensate la fatica di scrivere 50 linee come questa!

Invece, potete usare un array cui assegnare i valori...

```
10 DIM A(50)
20 FOR Z=1 TO 50
30 LET A(Z)=Z ←
40 NEXT Z
```

← Questa istruzione attribuisce i valori di Z all'array A, partendo da 1 e finendo a 50.

Mandate in esecuzione il programma; quindi dite al vostro computer: PRINT A(37) RETURN. Il calcolatore ha memorizzato 37 in questo elemento dell'array; e perciò stamperà 37, finché voi non cambierete A(37) oppure spegnerete il computer (calcellando così A(37)).

Se digitate...

```
PRINT A(5)+A(9)
```

avrete in risposta 14 finché non modificherete questi valori o li cancellerete dalla memoria.

E con gli **ARRAY** potete suonare un po' di musica.

```
10 REM SUONATA DI ARRAY... 1
20 DIM A(50),B(50),C(50),D(50)
30 FOR Z=1 TO 50
40 LET A(Z)=0:LET B(Z)=0:LET C(Z)=0:LET D(Z)=0:REM AZZERAMENTO INIZIALE
50 LET A(Z)=Z*5:LET B(Z)=Z*4:LET C(Z)=Z*3:LET D(Z)=Z*2
60 NEXT Z
70 FOR N=1 TO 50
80 SOUND 0,A(N),10,10:SOUND 1,B(N),10,10:SOUND 2,C(N),10,10:SOUND 3,D(N),10,10
90 NEXT N
100 FOR N=50 TO 1 STEP -1
110 SOUND 0,A(N),10,10:SOUND 1,B(N),10,10:SOUND 2,C(N),10,10:SOUND 3,D(N),10,10
120 NEXT N
130 GOTO 70
```



Cambiando poche linee,
potete avere la visualizzazione sullo
schermo di tutti i valori degli ARRAY.

```
10 REM SUONATA DI ARRAY... 2
20 DIM A(50),B(50)
30 FOR Z=1 TO 50
40 LET A(Z)=0:LET B(Z)=0:REM AZZERAMENTO INIZIALE
50 LET A(Z)=Z*5:LET B(Z)=Z*4
60 NEXT Z
70 FOR N=1 TO 50
80 SOUND 0,A(N),10,10:SOUND 1,B(N),10,10
90 NEXT N
100 FOR N=50 TO 1 STEP -1
110 SOUND 0,A(N),10,10:SOUND 1,B(N),10,10
120 NEXT N
130 SOUND 0,0,0,0:SOUND 1,0,0,0
140 FOR N=1 TO 50:PRINT "A(";N;")=";A(N):SOUND 1,A(N),
10,10:NEXT N
150 FOR N=1 TO 50:PRINT ,"B(";N;")=";B(N):SOUND 1,B(N),
10,10:NEXT N
```

Come già specificato, sugli elementi di un array potete anche eseguire operazioni matematiche.

Se mandate in esecuzione il programma precedente e premete SYSTEM RESET, i valori negli elementi degli ARRAY si conservano.

Per cui, digitando... PRINT A(50) RETURN
...dovreste ottenere 250.

Ora, digitate... PRINT B(50) RETURN

...e otterrete 200.

Infine, digitando... PRINT A(50) + B(50) RETURN

...dovreste ottenere 450.

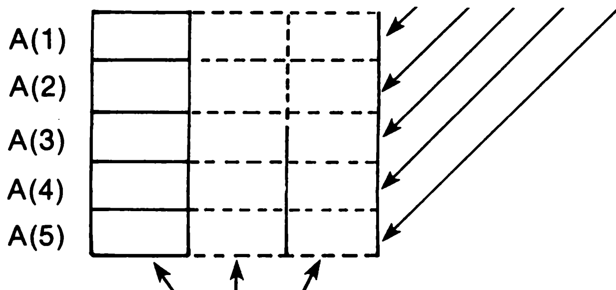
Gli ARRAY sono utili in programmi che trattano e manipolano grosse quantità di numeri:

- le letture di 40 strumenti di misura potrebbero essere registrate in un array da S(1) a S(40);
- le precipitazioni medie di pioggia nelle 100 più grandi città europee potrebbero stare da P(1) a P(100);
- i tassi di produzione giornaliera di hamburger da parte dei vostri 175 punti di distribuzione potrebbero essere memorizzati in un array da H(1) a H(175).

Il rompicapo delle MATRICI

Una versione più potente e complessa di un ARRAY è detta MATRICE.

Una MATRICE non solo ha delle *righe* di variabili che si susseguono verso il basso...

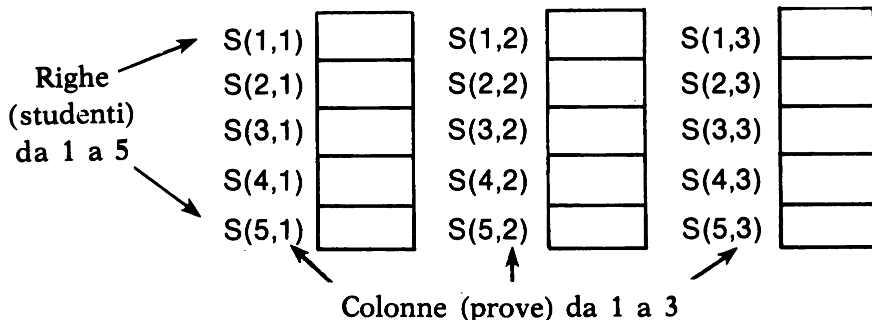


...ma anche delle *colonne* che si sviluppano per traverso.

Nel nostro esempio, supporremo una classe con solo 5 studenti (gli altri si sono stancati e se ne sono andati).

Ogni studente esegue tre prove: il risultato della prima prova figura a colonna 1; quello della seconda prova, a colonna 2; e così via. Per scrupolo di imparzialità, si fa riferimento ad ogni studente con una S seguita da un numero.

Perciò, la matrice dei punteggi delle prove degli studenti appare come segue...



Colonne (prove) da 1 a 3

Quello che ancora manca nella matrice sono i punteggi effettivi degli studenti.

Benché sia possibile assegnarli in questo modo...

LET S(3,1) = 78

...useremo nel programma l'istruzione INPUT.

OSSERVAZIONE: Una volta di più, è necessario eseguire il DI-Mensionamento della matrice che si attua col comando:,

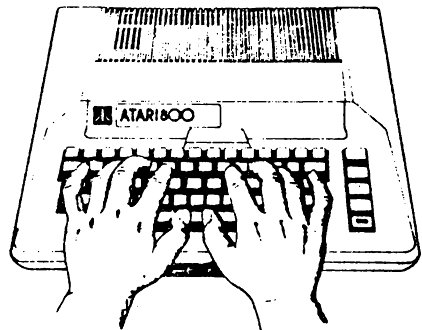
Questo è il numero di righe

DIM S(5,3)

Questo è il numero di colonne

Se battete NEW e quindi digitate il programma, scoprirete che la bellezza della MATRICE consiste nella possibilità di manipolare sia le righe sia le colonne. Oppure saltate questo pezzo e andate direttamente al programma successivo, che è più complesso.

```
10 DIM S(5,3)
20 FOR COL=1 TO 3
30 FOR RIG=1 TO 5
40 PRINT "VOTO DELLO STUDENTE N.";
RIG;" PROVA N.";COL
50 INPUT VOTO
60 S(RIG,COL)=VOTO
70 NEXT RIG
80 NEXT COL
90 PRINT " "
100 FOR X=1 TO 3
110 FOR Z=1 TO 5
120 IF X=1 THEN POSITION 2,Z
130 IF X=2 THEN POSITION 15,Z
140 IF X=3 THEN POSITION 28,Z
150 PRINT "S(";Z;",";X;")=";S(Z,X)
160 NEXT Z
170 NEXT X
```

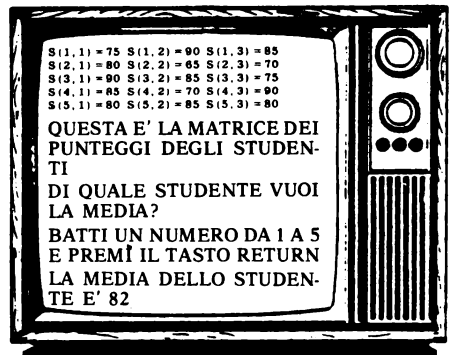


NOTA: Prometto che questo è il programma più lungo che mai vi chiederò di digitare sul vostro computer.



```
10 REM PROGRAMMA STUDENTI
20 PRINT "}"
30 REM MATRICE 1
40 DIM S(5,3)
50 FOR COL=1 TO 3
60 FOR RIG=1 TO 5
70 POSITION 2,11
80 PRINT "VOTO DELLO STUDENTE N.";RIG;" ";
90 IF COL=1 THEN PRINT "NELLA PRIMA PROVA"
100 IF COL=2 THEN PRINT "NELLA SECONDA PROVA"
110 IF COL=3 THEN PRINT "NELLA TERZA PROVA"
120 INPUT VOTO:PRINT " "
130 IF VOTO>100 THEN PRINT "FURBETTO,
IL MASSIMO E´ 100! RIPETI.";:PRINT :PRINT
:GOTO 120
140 S(RIG,COL)=VOTO
150 NEXT RIG
160 NEXT COL
170 PRINT "}"
180 FOR X=1 TO 3
190 FOR Z=1 TO 5
200 IF X=1 THEN POSITION 2,Z
210 IF X=2 THEN POSITION 15,Z
220 IF X=3 THEN POSITION 28,Z
230 PRINT "S(";Z;",";X;")=";S(Z,X)
240 NEXT Z
250 NEXT X
260 POSITION 1,8:PRINT "QUESTA E´ LA MATRICE DEI PUNTEGGI DEGLI STUDENTI"
270 POSITION 1,10:PRINT "DI QUALE STUDENTE VUOI LA MEDIA?"
280 PRINT :PRINT "BATTI UN NUMERO DA 1 A 5 E PREMI IL TASTO RETURN"
290 INPUT S
300 IF S>5 THEN PRINT " ":POSITION 1,1:PRINT "CI SONO SOLO 5 STUDENTI":FOR RIT=1
 TO 300:NEXT RIT:GOTO 170
310 PRINT :PRINT "LA MEDIA DI QUESTO STUDENTE E´ ";INT((S(S,1)+S(S,2)+S(S,3))/3)
320 GOTO 270
```

Al termine il vostro schermo dovrebbe apparire così



Ovviamente i valori dipenderanno dai dati introdotti

Il computer potrebbe anche calcolare facilmente la media dei punteggi per ogni prova; aggiungete semplicemente le linee che gli dicono di sommare i 5 punteggi di ogni colonna e di dividere per 5.

C'è una quantità di altre applicazioni per le MATRICI e sono molti gli esempi che potete trovare in altri libri di Basic. Generalmente, quasi tutto ciò che può essere disposto secondo righe e colonne può essere organizzato in una MATRICE.

Per esempio, un Vice Presidente alle Vendite potrebbe creare un proprio rapporto mensile di aspetto simile a questo...

Rapporto Mensile
dall'ufficio di U. Rossi
Vendite in milioni per Prodotto e Zona

	trottole	trappole	racchette	rocchetti	milioni per zona
Zona n. 1					
Zona n. 2					
Zona n. 3					
Zona n. 4					
Zona n. 5					
TOTALI					
	totale trottole	totale trappole	totale racchette	totale rocchetti	totale vendite

Parte 2

Grafica, Colore e Suono

Generalità sulla Grafica ATARI

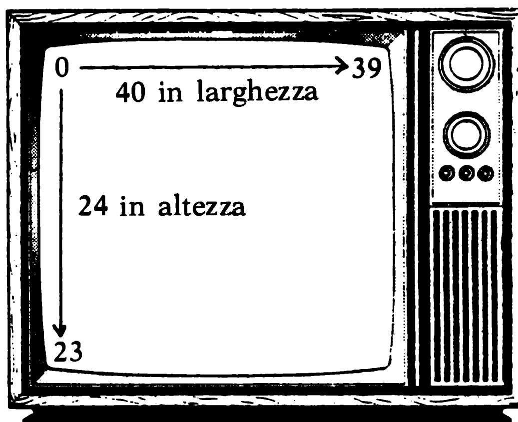
Gli Home Computer ATARI hanno delle possibilità grafiche veramente notevoli.
Alcuni aspetti avanzati della grafica richiedono ben più di una conoscenza introduttiva sulla programmazione e il progetto degli elaboratori; tuttavia questo capitolo intende avviarvi alla grafica col Computer ATARI.

I Computer ATARI hanno *3 modi "testo"* per rappresentare parole sullo schermo. Il primo è il...

MODO GRAFICO 0
(schermo blu normale)

Si entra in questo modo:

- all'accensione del computer
- premendo SYSTEM RESET
- digitando GR.0 RETURN



GR.0 per caratteri ad altezza normale

Nel modo GRAFICO 0 potete posizionarvi da 0 a 39 in orizzontale e da 0 a 23 in verticale. Il computer normalmente si sposta di due spazi a destra quando scrive sullo schermo, ma non lo fa con la stampante.

Utilizzando l'istruzione POSITION nel modo grafico 0 potete scrivere in ogni punto dello schermo.

posizione orizzontale **posizione verticale**

```
10 POSITION 16,11
20 PRINT "CENTRO"
```

Potete impiegare POSITION ad ogni comando di stampa, onde mantenere le uscite tutte su una stessa porzione dello schermo.

Eseguite dapprima questo programma:

```
10 PRINT "↰":POKE 752,1
20 POSITION 19,11
30 PRINT X
40 X=X+1
50 GOTO 20
```

Quindi modificate la linea 50...

50 GOTO 30

e osservate la differenza.

POSITION può essere utilizzato per enfatizzare un messaggio conferendogli una sorta di movimento.

Provate il programma...

```
10 PRINT "↰":POKE 752,1
20 FOR X=1 TO 16
30 POSITION X,11
40 PRINT "CENTRO"
50 FOR RIT=1 TO 30:NEXT RIT
60 FOR Y=0 TO 15
70 POSITION Y,11:PRINT " "
80 NEXT Y
90 NEXT X
```

oppure una versione leggermente più raffinata che vede i caratteri camminare attraverso lo schermo.

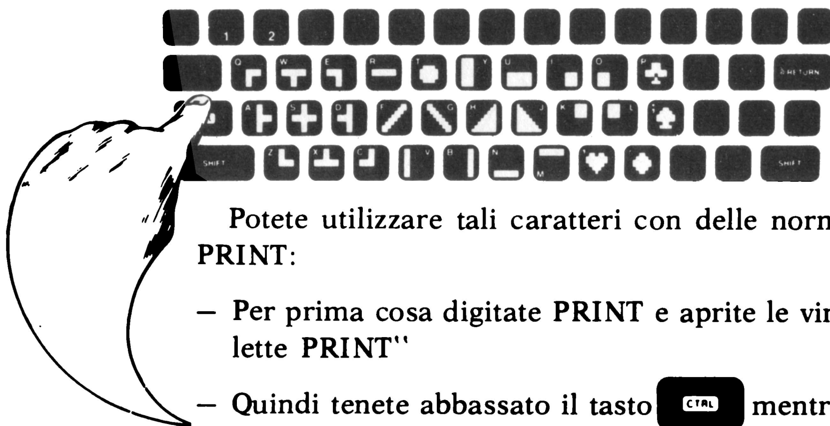
```
10 PRINT "↰":POKE 752,1
20 FOR X=1 TO 16
30 POSITION X,11
40 IF X/2=INT(X/2) THEN POSITION
X,11:PRINT "C/NT/O":GOTO 60
50 IF X<16 THEN POSITION X,11:PRINT
"C/NT/O"
60 IF X=16 THEN POSITION X,11:PRINT
"CENTRO"
70 FOR RIT=1 TO 50:NEXT RIT
80 FOR Y=0 TO 15
90 POSITION Y,11:PRINT " "
100 NEXT Y
110 NEXT X
```

Errori semplici ma comuni, frequentemente commessi mentre s'impara ad usare l'istruzione POSITION.

1. Posizionamento fuori schermo
   Esempio: POS. 40,12
   nel modo grafico 0, 39 è il numero massimo in orizzontale e 23 il numero massimo in verticale.
2. Abbreviazione POS. senza il punto.
3. Scambio della O di POS con uno zero.

Anche in Modo Grafico 0 si usano Caratteri Grafici

Probabilmente avete già incontrato, intenzionalmente o per caso, i caratteri grafici che sono disponibili per voi anche nel modo grafico 0.



Potete utilizzare tali caratteri con delle normali PRINT:

- Per prima cosa digitate PRINT e aprite le virgolette PRINT"
- Quindi tenete abbassato il tasto CTRL mentre...
- Battete uno dei caratteri illustrati sopra.
- Infine chiudete le virgolette e premete il tasto RETURN

Per disegnare una finestra, per esempio, battete questo programma.
Tenete premuto il tasto CTRL mentre digitate queste lettere

```
10 PRINT "QWE"     sul video si      10 PRINT "┌──┬──┐"
20 PRINT "ASD"     presenterà    →   20 PRINT "├──┼──┤"
30 PRINT "ZXC"     così              30 PRINT "└──┴──┘"
```

Mandate in esecuzione il programma e successivamente aggiungete...

```
40 GOTO 10
```

Si produrranno le finestre di un altissimo grattacielo.

Per disegnare rettangoli con i caratteri grafici, sperimentate le combinazioni seguenti di tasti.

Tenete premuto il tasto CTRL mentre digitate questi caratteri

10 PRINT "QRRRRRRE" che appare "┌──────┐"
20 PRINT "| |"

Questi sono stampati premendo insieme i tasti SHIFT e ↓; sono i fianchi del rettangolo.

IMPORTANTE... Come realizzare i fianchi del rettangolo.

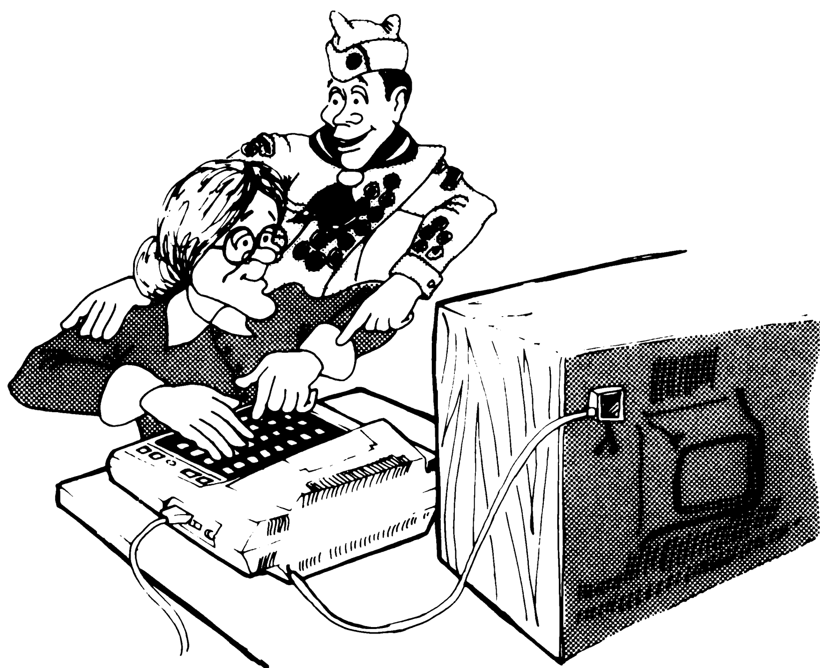
30 PRINT "ZRRRRRRC" che appare così "└──────┘"

Tenete premuto il tasto CTRL mentre digitate questi caratteri.

Eseguite il programma e provate anche il seguente, per una maggior flessibilità sulle dimensioni dei rettangoli.

```
10 PRINT "ALTEZZA RETTANGOLO";
20 INPUT H
30 PRINT "┌──────┐"
40 FOR ALT=1 TO H
50 PRINT "|      |"
60 NEXT ALT
70 PRINT "└──────┘"
80 FOR RIT=1 TO 200:NEXT RIT
90 RUN
```

Suggerimenti Utili per l'uso dei caratteri grafici



– I tasti CTRL T producono un bel ● nel caso vogliate disegnare un grosso punto.

– Se volete far delle prove con i caratteri grafici, potete premere insieme i tasti CTRL e CAPS LOWR, fissando così la tastiera in *modalità "grafica a caratteri";* per tornare al modo normale, premete SYSTEM RESET o SHIFT e CAPS LOWR insieme.

– La stampante non può riprodurre i caratteri grafici; la grafica a caratteri è prerogativa esclusiva delle rappresentazioni video.

– Ricordate che, purché compresi tra le virgolette, potete utilizzare operatori matematici, segni di punteggiatura e caratteri in *inverse video* per produrre le immagini.

Ecco un semplice programma che usa i simboli

MINORE DI <

e

MAGGIORE DI >

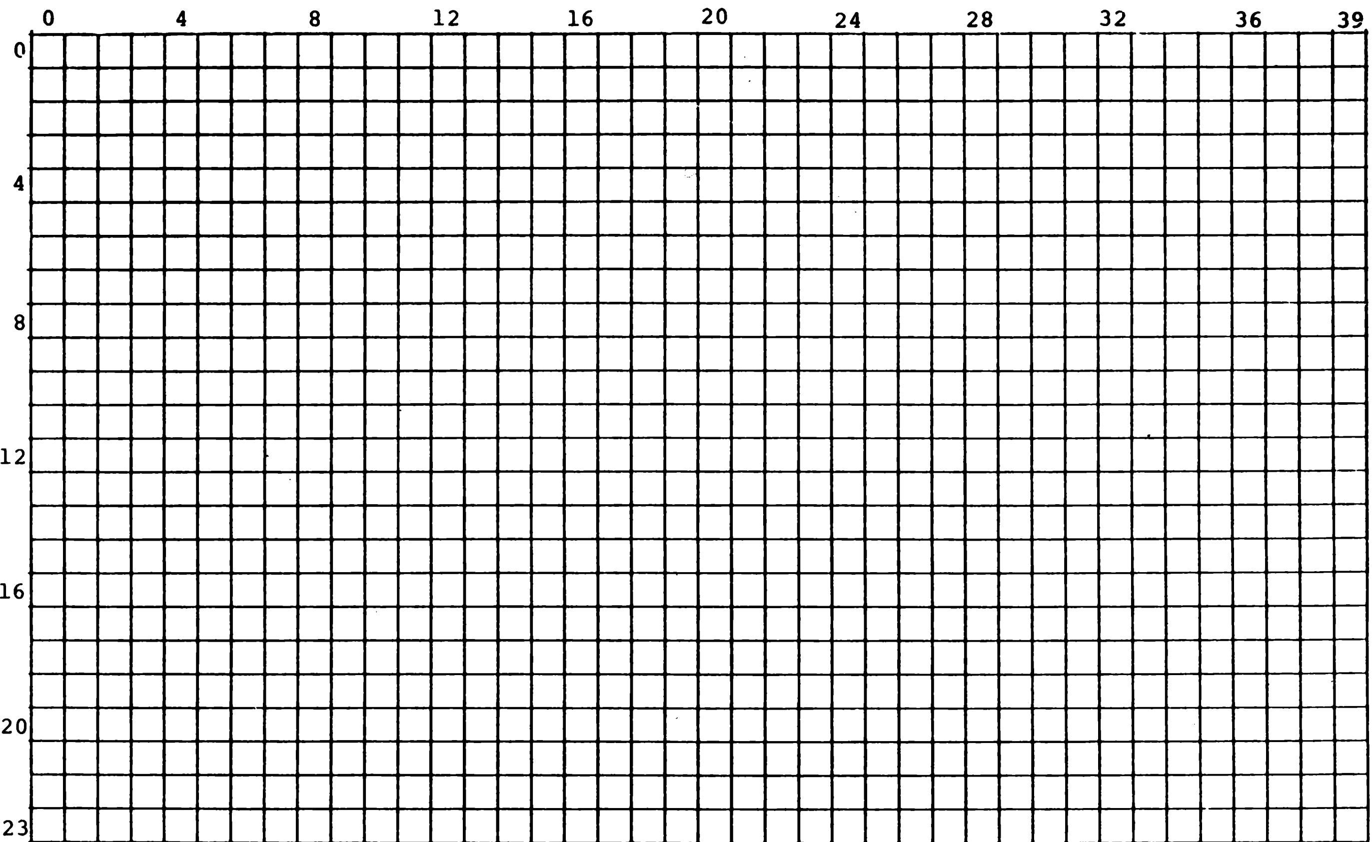
per dare l'illusione del movimento.

```
5 PRINT "↖":POKE 752,1
10 POSITION 19,12:PRINT "<>"
20 FOR RIT=1 TO 50:NEXT RIT
30 POSITION 19,12:PRINT "><"
40 FOR RIT=1 TO 50:NEXT RIT
50 GOTO 10
```



Il tracciato seguente può essere impiegato per predisporre le immagini video nel modo grafico 0.

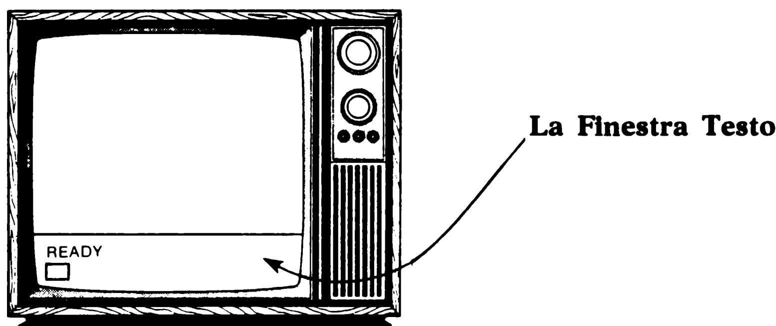
Modo Grafico 0



Note: ____________________

Modo Grafico 1: per testi PIÙ GRANDI

Il modo che viene dopo GR.0 è GR.1; lo si ottiene premendo GR.1 RETURN. Come potete osservare, siete stati portati verso il basso dello schermo e avete scoperto la "FINESTRA TESTO".



La finestra testo è un vero godere perchè, a differenza dei parenti acquisiti, può rivelarsi assai utile e può essere allontanata in fretta e facilmente!

Potete scrivere una linea ad altezza normale sulla riga più alta della finestra testo.

Provate...

PRINT "CIAO FINESTRA"



oppure potete utilizzare *il 6 magico*, e scoprire che cosa succede...

PRINT # 6; "CIAO SCHERMO"

Il vostro video dovrebbe apparire così



Se non succede, probabilmente avete dimeticato qualcosa... Non preoccupatevi; andate nel modo testo successivo ed arriverete rapidamente allo stesso risultato.

Modo Grafico 2: per testi ANCOR PIÙ GRANDI

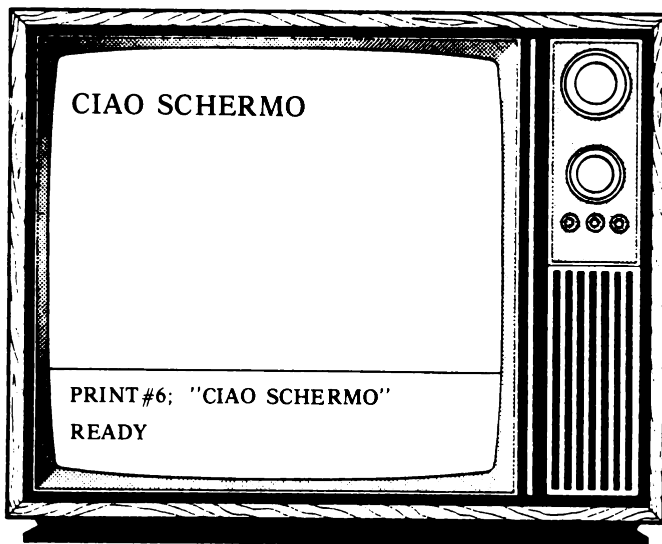
Digitate...

GR. 2 RETURN
PRINT "CIAO FINESTRA"

Ora provate...

PRINT # 6; "CIAO SCHERMO"

Il video dovrebbe apparire come segue...



*** Ancora Complicazioni ***

Scrivete...

PRINT # 6; " CIAO SCHERMO "

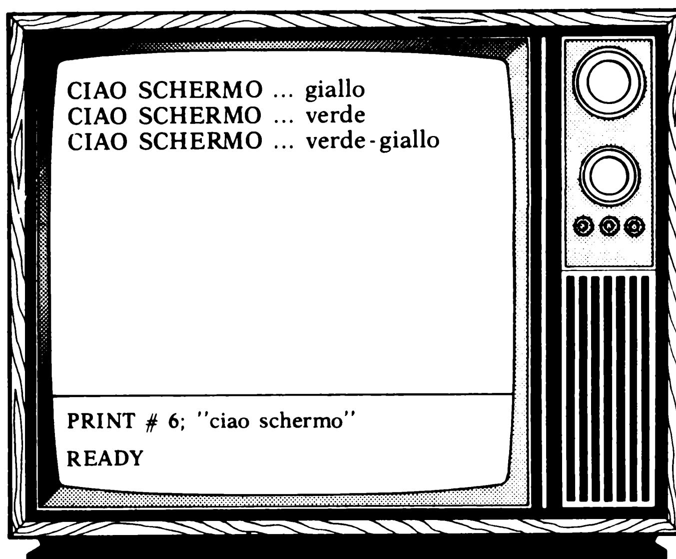
(premete il tasto in questi punti per entrare e uscire dall'INVERSE VIDEO).

ora provate...

PRINT # 6;"ciao schermo"

(premete qui il tasto CAPS LOWR e qui i tasti CAPS LOWR SHIFT per entrare e uscire dal minuscolo).

Se è andato tutto bene e il colore del televisore è ben regolato, il video dovrebbe apparire così...



Se non compare nulla di tutto ciò, premete SYSTEM RESET e tornate al punto in cui avete digitato GR.2 RETURN.

Non oltrepassate questo capitolo senza averlo ben appreso, o non verrete più a capo di nulla.

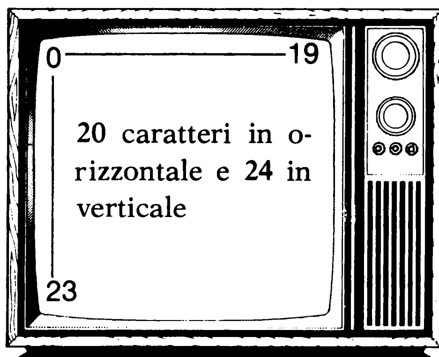
NOTA: Combinando il minuscolo con l'inverse video, potreste far comparire un ulteriore colore in più; il capitolo sul COLORE e sulle PEEK e POKE vi spiegherà come tingere del colore desiderato ogni vostro testo.

Uso di POSITION nei Modi Grafici 1 e 2

Potete utilizzare il comando POSITION nei modi grafici 1 e 2, ma dovete tener conto del minor numero di punti. Se ben ricordate, nel modo 0 ci sono 40 spazi in orizzontale e 24 in verticale.

Nel modo 1, disponete di...





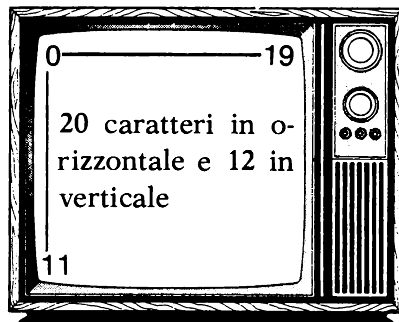
oppure senza la finestra testo

GR.1: POS. 17,11: PRINT "CIAO!"

collocherà la scritta "CIAO" a metà schermo e tutta a destra.

Nel modo 2, avete spazio per...





oppure senza la finestra testo

Ricordate, quando nei modi grafici 1 e 2 volete visualizzare testi maggiorati, dovete adottare un # 6; tra il **PRINT** e le virgolette iniziali. Senza il # 6; finireste per stampare nella finestra testo sul basso dello schermo.

```
10 GRAPHICS 2
20 PRINT #6;"TESTO MAGGIORATO"
30 POKE 752,1:REM DISATTIVA CURSORE
40 PRINT "TESTO PICCOLO, MA BUONO UGUALMENTE"
50 FOR RIT=1 TO 300:NEXT RIT
60 PRINT
70 FOR RIT=1 TO 300:NEXT RIT
80 GOTO 40
```

Eliminare la Finestra Testo

** può essere fatto solo da programma **

Per *far fuori* la finestra testo, aggiungete 16 al numero del modo grafico, qualunque esso sia.

+16 vuol dire ''no finestra testo''

Digitate... NEW RETURN

```
10 GRAPHICS 2+16
20 PRINT #6;"TESTO MAGGIORATO"
30 PRINT #6;"SENZA FINESTRA TESTO"
40 GOTO 40:REM QUESTA LINEA MANTIENE LA SCRITTA SUL VIDEO
```

NOTA: Potreste vedere i listati di qualcuno che scrive...

10 GR. 18

che è lo stesso di GR. 2+16. Tuttavia sembra altrettanto comodo e più chiaro utilizzare il numero del modo grafico standard e sommargli 16.

Eccovi un programma *scenografico* che combina i modi di GR.0 e GR.1 con qualche altro trucchetto...

```
10 PRINT "↰":DIM NOME$(20):POSITION 2,7
20 PRINT "SCRIVI IL TUO NOME E PREMI IL TASTO RETURN"
30 POSITION 4,12:PRINT "NOME";:INPUT NOME$
40 GRAPHICS 2+16
50 POSITION (20-LEN(NOME$))/2,4:REM CENTRA IL NOME
60 PRINT #6;NOME$
70 POSITION 1,7:PRINT #6;"E´ UN SOGGETTO INTERESSANTE":REM QUESTO IN INVERSE VID
EO
80 FOR FLIP=0 TO 20
90 FOR FLASH=0 TO 14
100 SETCOLOR 0,0,FLASH:SOUND 0,FLASH,10,10
110 NEXT FLASH
120 FOR RIT=1 TO 20:NEXT RIT
130 NEXT FLIP
140 SETCOLOR 0,0,14:SOUND 0,0,0,0
150 FOR RIT=1 TO 1000:NEXT RIT
160 SETCOLOR 0,0,0:SETCOLOR 2,0,0
170 FOR RIT=1 TO 800:NEXT RIT
180 RUN
```

NOTA: Potete salvare la linea 50, per centrare in modo GR.2 ogni input di nome NOME$

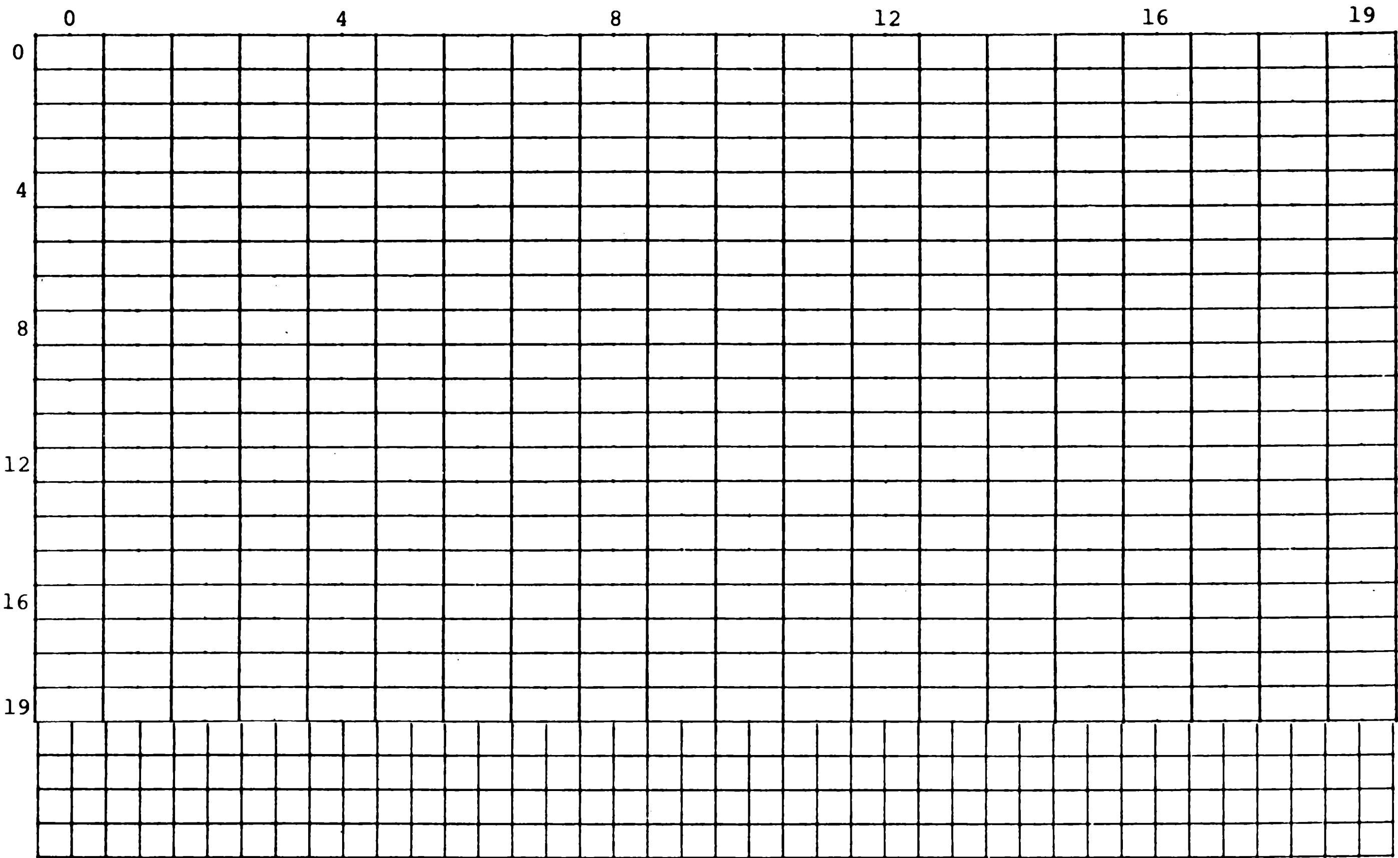
e

salvare le linee da 80 a 130 per far lampeggiare le scritte, con suono.

Nel capitolo sul COLORE potete trovare ulteriori dettagli circa il funzionamento di programmi di questo tipo e il significato della SETCOLOR.

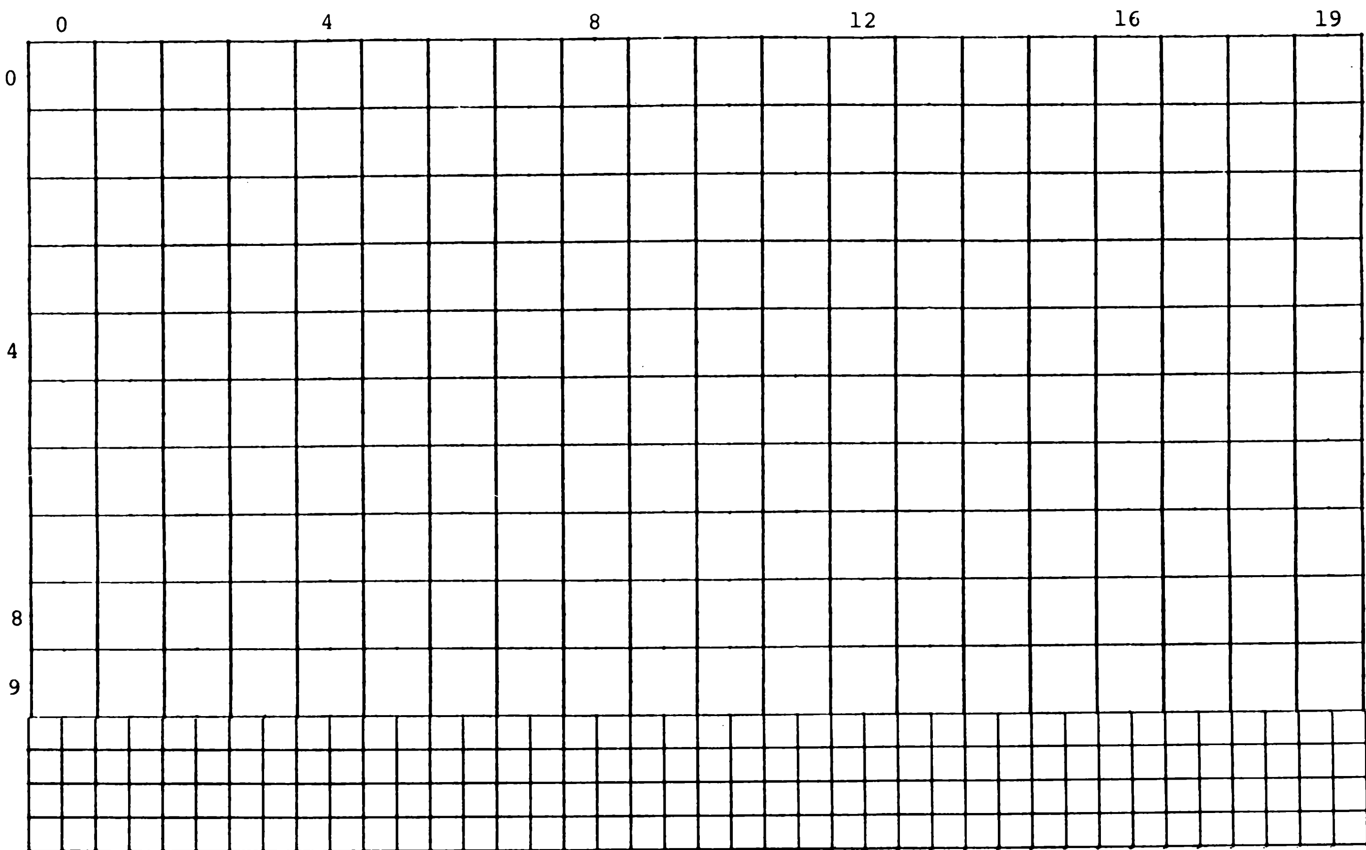
Le quattro pagine seguenti vi aiuteranno a predisporre le immagini nei modi GR.1 e GR.2, con o senza finestra testo. Osservate che, benchè la larghezza sia la stessa, i caratteri del modo grafico 1 sono alti la metà di quelli del modo 2.

Modo Grafico 1 con Finestra Testo

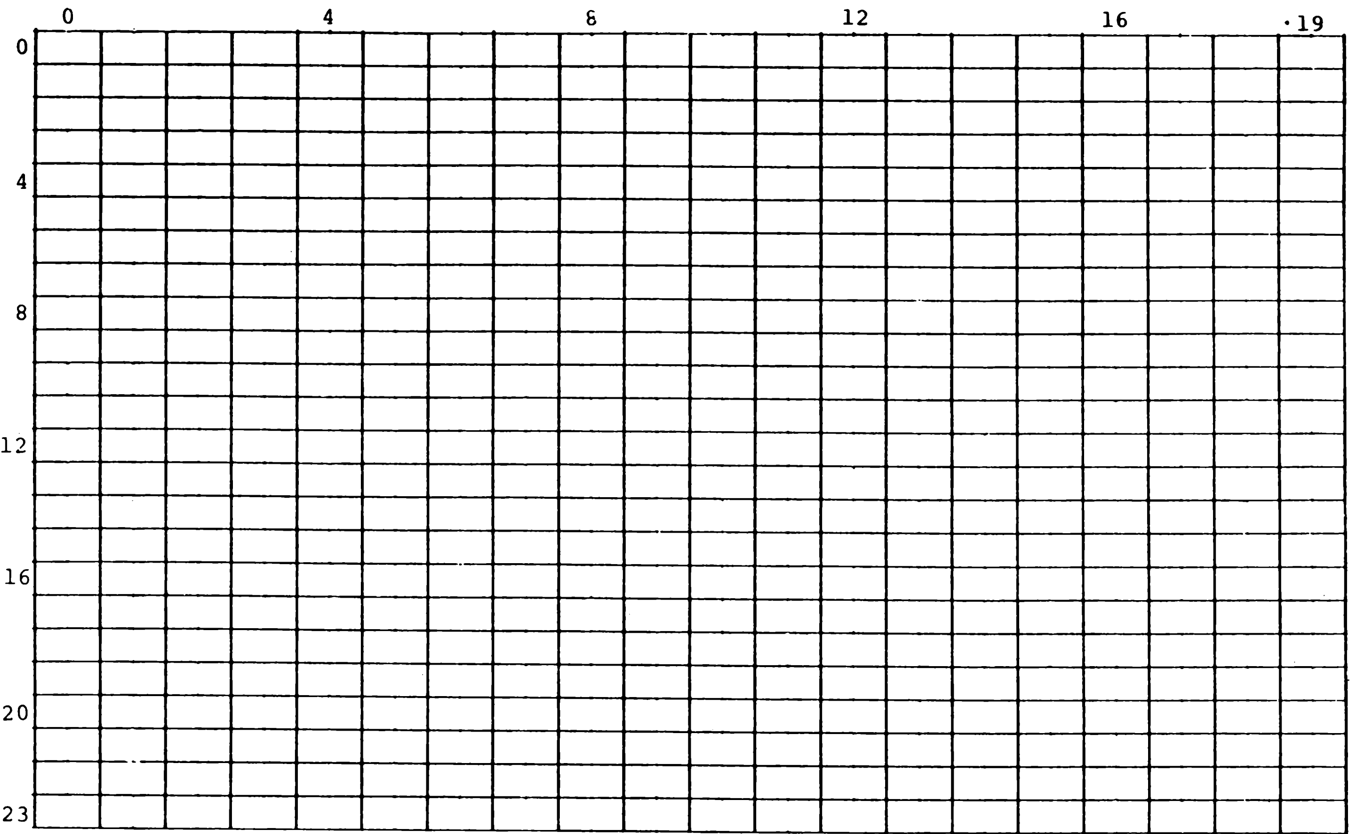


Modo Grafico 2

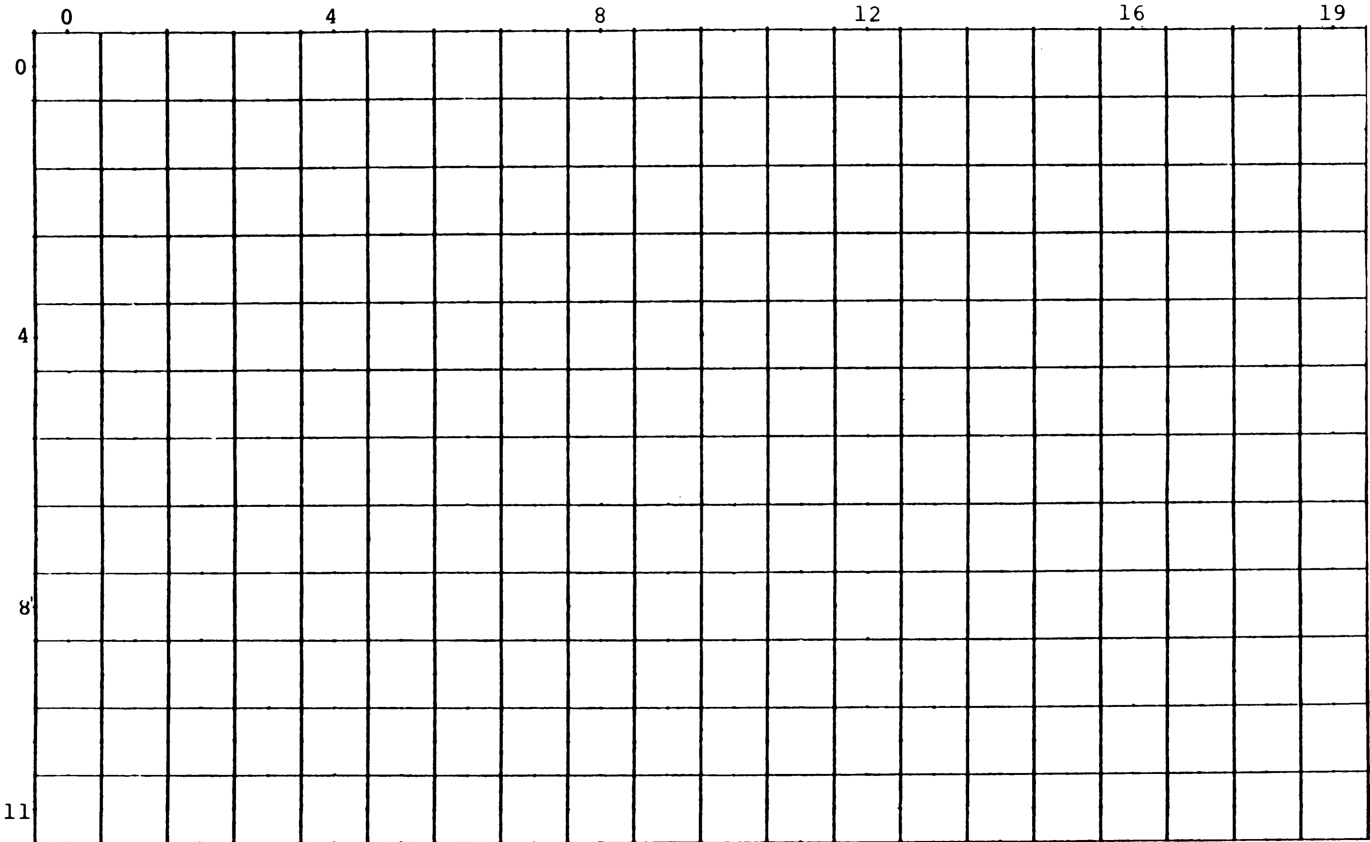
con Finestra Testo



Modo Grafico 1

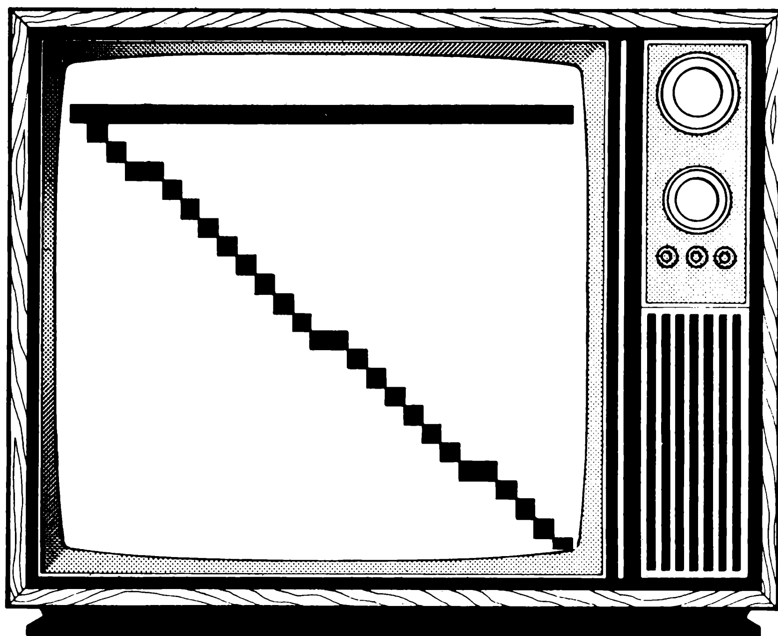


Modo Grafico 2



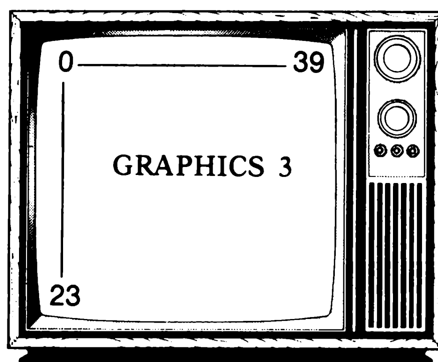
Modi Grafici da 3 a 8

Sono questi i modi propriamente *grafici,* poiché abitualmente in essi si rappresentano blocchi colorati invece di lettere o parole.



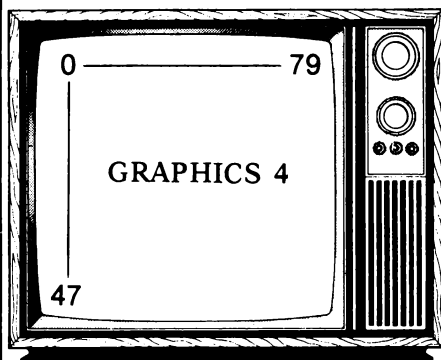
Come regola generale, andando verso i numeri di modo più elevati, disporrete di un maggior numero di punti (detti *pixel*), con la possibilità di disegni più dettagliati.

Ovviamente questo criterio è troppo elementare per un progettista di computer: per cui, eccovi le idee fondamentali sui *"traffici"* da fare per utilizzare i singoli modi grafici...

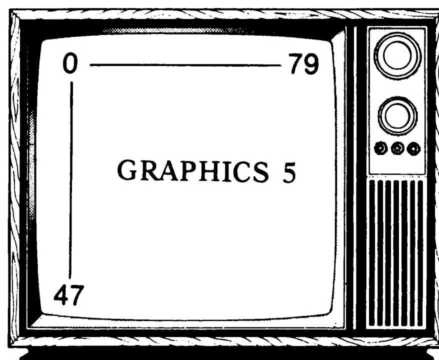


GR.3 ha la stessa risoluzione di GR.0, ma impiega blocchi di colore invece di caratteri. Con questo modo non potete ottenere disegni troppo dettagliati.

I due modi seguenti hanno la stessa risoluzione.

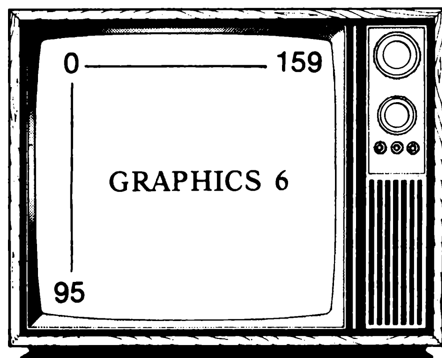


Questo impiega pochi colori (2) ma anche meno memoria; utilizzatelo se dovete risparmiare la memoria.

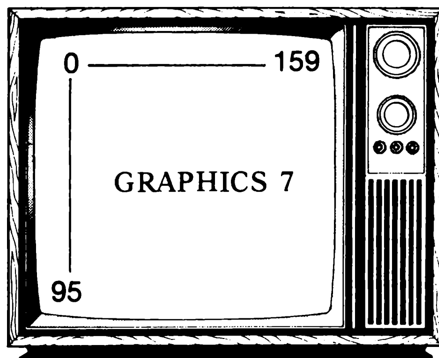


questo ha più colori ma impiega il doppio di memoria.

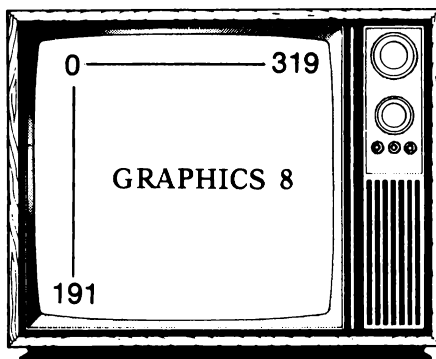
I due seguenti hanno la stessa risoluzione, moderatamente elevata.



come prima, metà colori ma anche metà memoria impegnata.

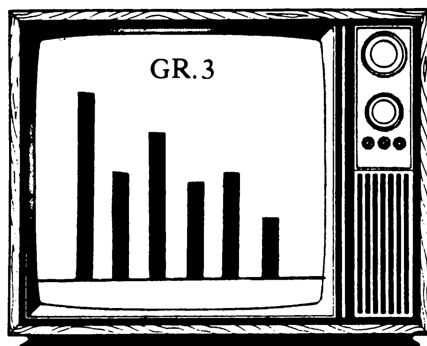


questo ha una risoluzione piuttosto buona e dispone di 4 colori. Tuttavia, occupa il doppio di memoria rispetto a GRAPHICS 6.

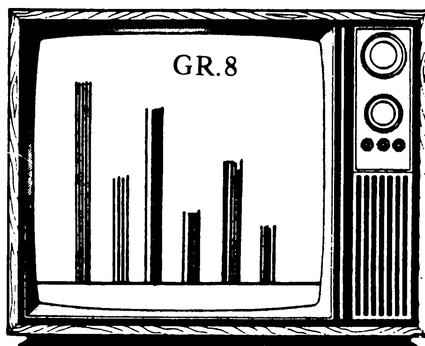


Risoluzione molto elevata, 2 colori, un mare di memoria usata. Utilizzatelo quando sono importanti i dettagli grafici.

Come regola generale, esaminate che tipo di rappresentazione vi serve e rendete la vita facile a voi e al computer. Per esempio, nel disegnare un *diagramma a barre*, è più probabile che utilizziate GR.3 piuttosto che GR.8.

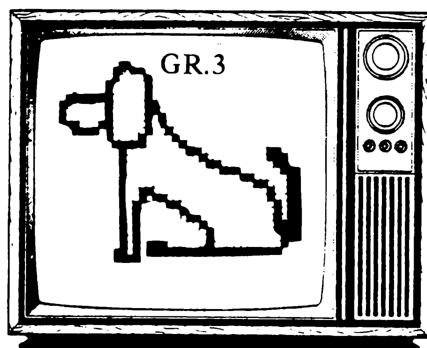


Abbondanti colori; non molta memoria; e non troppi punti da tracciare.

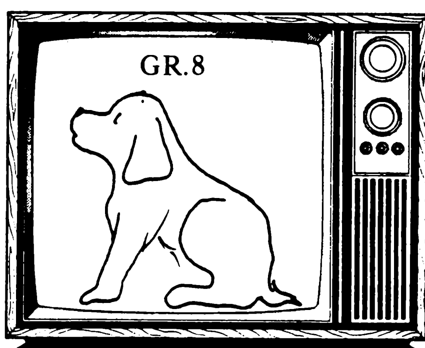


Si impiega solo un colore; occorre tracciare molte linee; maggior impiego di memoria.

D'altra parte, se volete disegnare uno *Springer Spaniel*...



Volete che questo cane vi porti le pantofole?



Un po' meglio, direi.

Riassumendo... Se volete disegnare un brutto diagramma a barre, utilizzare GRAPHICS 8; se volete disegnare un brutto cane, utilizzate GRAPHICS 3.

Tanto per gingillarvi...

Potete eseguire quanto segue in modo immediato, senza numeri di linea.

```
GR.3
COLOR 1
PLOT 2,2
DRAWTO 37,2

DRAWTO 37,18
DRAWTO 2,2
```

(seleziona colore)

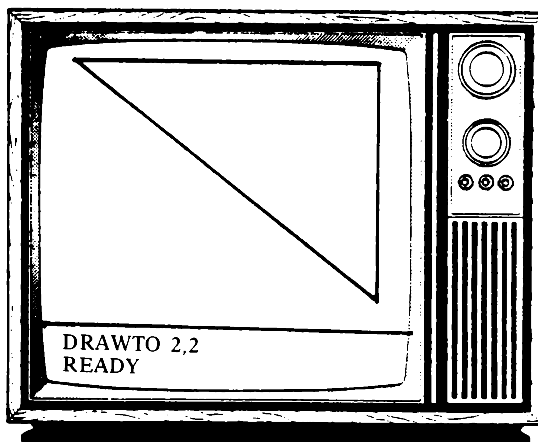
(PLOT significa emetti un punto colorato o inizia a disegnare da questo punto. È simile al POSITION dei modi testo)

(DRAWTO significa parti dalla posizione dell'ultima PLOT o DRAWTO e traccia fino qui una linea colorata)

Questa è la migliore linea diagonale che il computer possa rappresentare in GRAPHICS 3.

Per qualcosa di più raffinato...

```
GR.7
COLOR 1
PLOT 2,2
DRAWTO 150,2
DRAWTO 150,70
DRAWTO 2,2
```



Potete osservare che la diagonale adesso è molto più diritta.

E questo è il modo di massima definizione.

```
GRAPHICS 8
COLOR 1
PLOT 5,5
DRAWTO 300,5
DRAWTO 300,150
DRAWTO 5,5
```



Come vedete, più è alta la risoluzione, maggiore è il dettaglio raggiungibile.



Suggerimenti di Grafica ai principianti

Benché la Finestra Testo sia assai utile per far prove in modo immediato e per visualizzare senza difficoltà parole e grafici ad un tempo, spesso vorrete escluderla nei vostri programmi.

Perciò ricordate, da programma potete eliminarla aggiungendo 16 al numero del modo grafico

10 GR. 2+16 oppure 10 GR. 8+16

Utilizzano un modo grafico senza Finestra Testo, dovete anche ricordarvi di imporre al computer di mantenere l'immagine sullo schermo e di non tornare in condizione di READY.

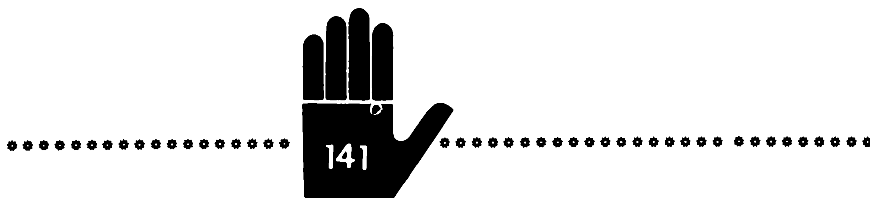
Un modo per ottenerlo è far sì che l'ultima linea del programma sia un *loop senza fine...*

Provate i seguenti programmi...

10 GRAPHICS 7 20 COLOR 1 30 PLOT 2,2:DRAWTO 150,80	Dopo l'esecuzione del programma, nella finestra testo ricompare "READY".
10 GRAPHICS 7+16 20 COLOR 1 30 PLOT 2,2:DRAWTO 150,80	Aggiungendo 16 al modo grafico, eliminate la finestra testo, ma il programma viene eseguito così in fretta che difficilmente riuscite a vederlo.
10 GRAPHICS 7+16 20 COLOR 1 30 PLOT 2,2:DRAWTO 150,80 40 GOTO 40	La linea 40 introduce un loop senza fine, che impedisce al programma di terminare.
10 GRAPHICS 7+16 20 COLOR 1 30 PLOT 2,2:DRAWTO 150,80 40 FOR RIT=1 TO 500:NEXT RIT	La linea 40, in nuova versione, produce un lieve ritardo prima della fine del programma.

Se volete per l'appunto costruire un'immagine per un breve intervallo di tempo e quindi andarvene in un'altra parte del programma, introducete un loop di ritardo nel programma stesso. Provate il seguente programma che alterna i modi 7 e 0.

```
10 PRINT "↖":PRINT "BATTI UN NUMERO TRA 1 E 800, QUINDI RETURN";:INPUT LUNG
20 FOR EX=1 TO 3
30 GRAPHICS 7:POKE 710,0
40 COLOR 1
50 PLOT 2,2:DRAWTO 158,2
60 DRAWTO 158,80
70 DRAWTO 2,2:POKE 752,1:PRINT "QUESTO E' IL PERIODO DI GR. 7"
80 FOR RIT=1 TO LUNG:NEXT RIT
90 GRAPHICS 0:POKE 752,1
100 SETCOLOR 2,8,6:SETCOLOR 1,0,12
110 POSITION 5,11
120 PRINT "QUESTO E' IL PERIODO DI GR. 0"
130 FOR RIT=1 TO LUNG:NEXT RIT
140 NEXT EX
150 RUN
```



Il messaggio d'ERRORE più frequente che incontrerete utilizzando modi grafici diversi è l'ERRORE 141. In parole povere vuol dire: "*adesso sei proprio andato fuori*"; ciò significa che vi siete portati per disegnare su un punto oltre i limiti di definizione del modo grafico che state adoperando. TORNATE INDIETRO!

****

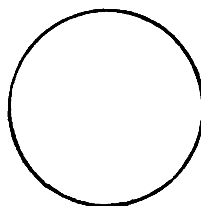
GR.3
COLOR 1
PLOT 45,2

Adesso dite al computer che vi dispiace e proponetegli...

PLOT 39,2

Poiché fare i conti per disegnare cerchi in Basic spesso implica più lavoro di quanto non appaia, ho incluso due *routine* che potete sperimentare. È forse una buona idea cominciare a raccogliere le routine che provate, per scegliere la più efficiente in relazione al programma che state scrivendo.

```
10 REM CIRCONFERENZA
20 GRAPHICS 7+16:COLOR 1
30 FOR IND=1 TO 400
40 A=A+0.05
50 X=SIN(A)*50:Y=COS(A)*45
60 PLOT X+80,Y+45
70 NEXT IND
80 FOR RIT=1 TO 500:NEXT RIT
```



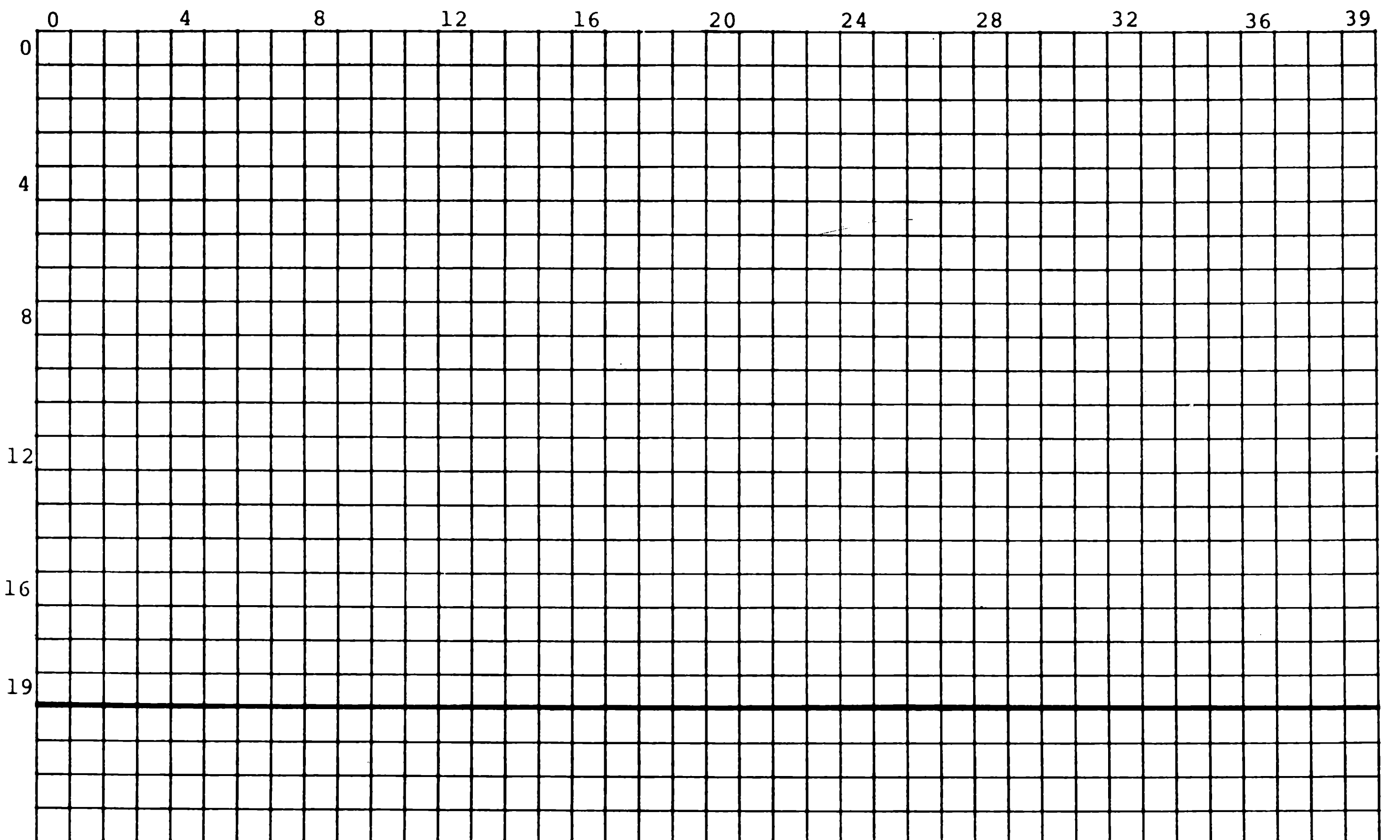
```
10 REM CIRCONFERENZA GRAPHICS 8
20 GRAPHICS 8+16:POKE 709,14:COLOR 1
30 FOR IND=1 TO 400
40 A=A+0.05
50 X=SIN(A)*50:Y=COS(A)*45
60 PLCT X+160,Y+85
70 NEXT IND
80 FOR RIT=1 TO 500:NEXT RIT
```

Con leggeri cambiamenti del programma precedente, si ottiene un'ellisse invece di una circonferenza.

```
10 REM ELLISSE GRAPHICS 8
20 GRAPHICS 8+16:POKE 709,14:COLOR 1
30 FOR IND=1 TO 400
40 A=A+0.05
50 X=SIN(A)*80:Y=COS(A)*15
60 PLOT X+160,Y+85
70 NEXT IND
80 FOR RIT=1 TO 500:NEXT RIT
```

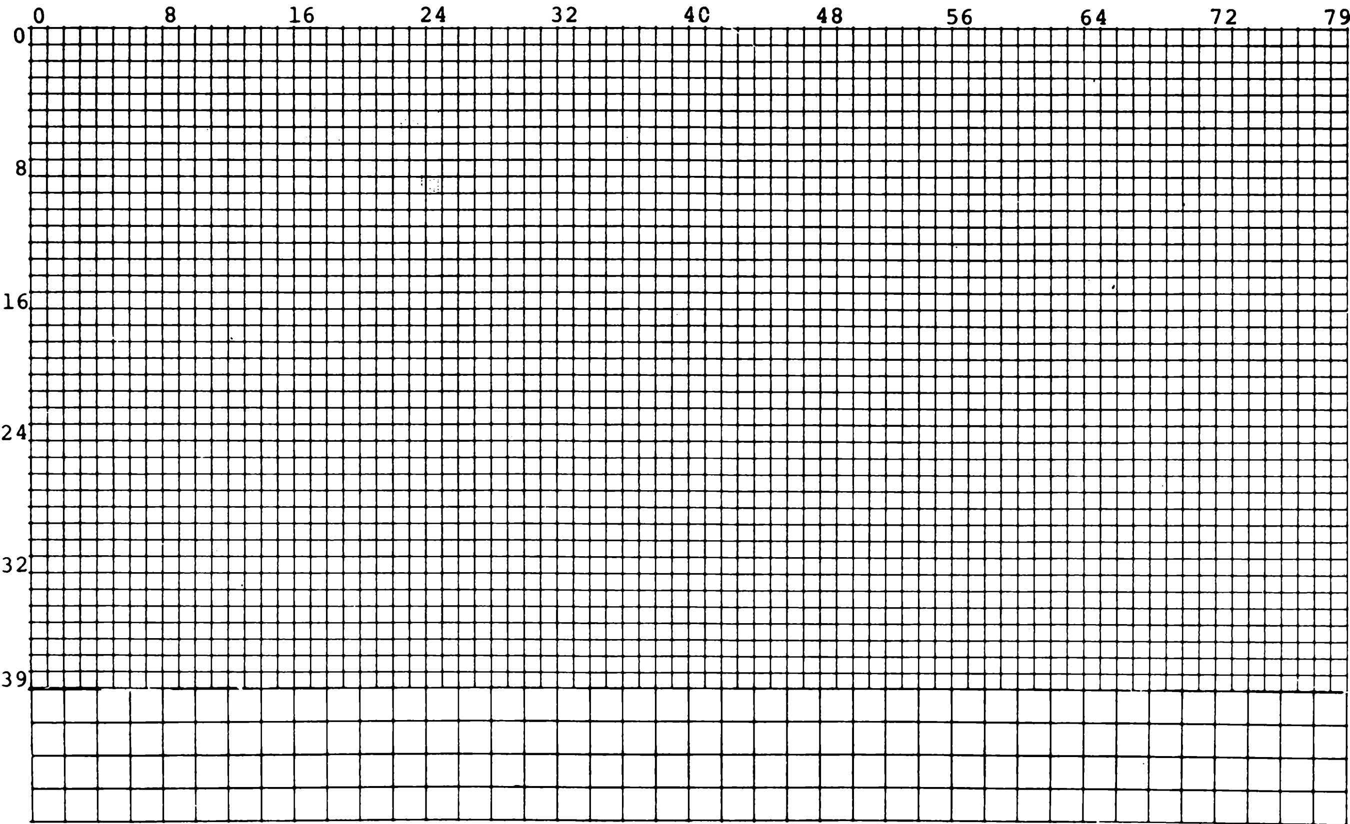
I moduli che seguono vi aiuteranno per progettare immagini nei modi grafici da 3 a 8.

Modo Grafico 3 con Finestra Testo



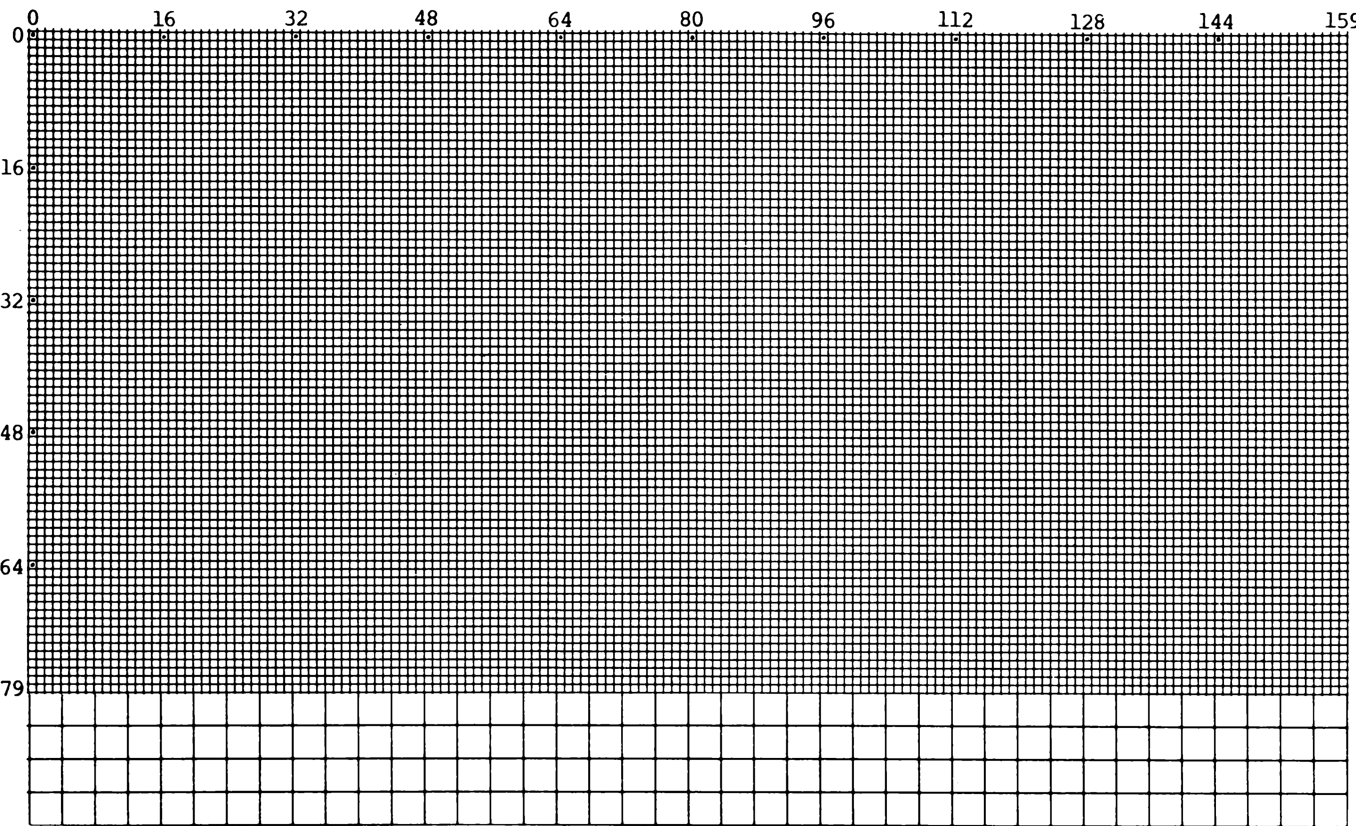
Modo Grafico 4 o 5

con Finestra Testo



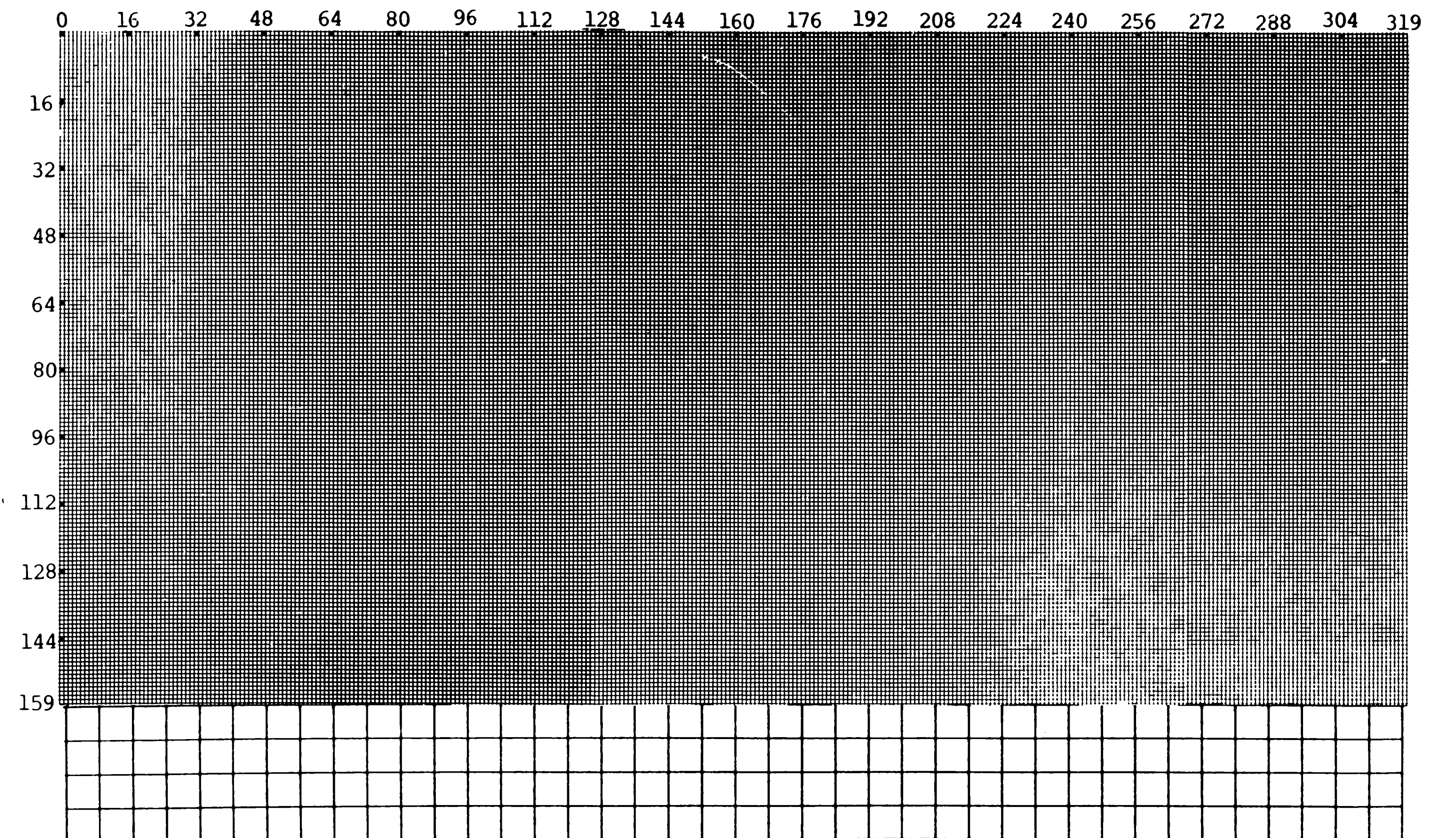
Modo Grafico 6 o 7

con Finestra Testo

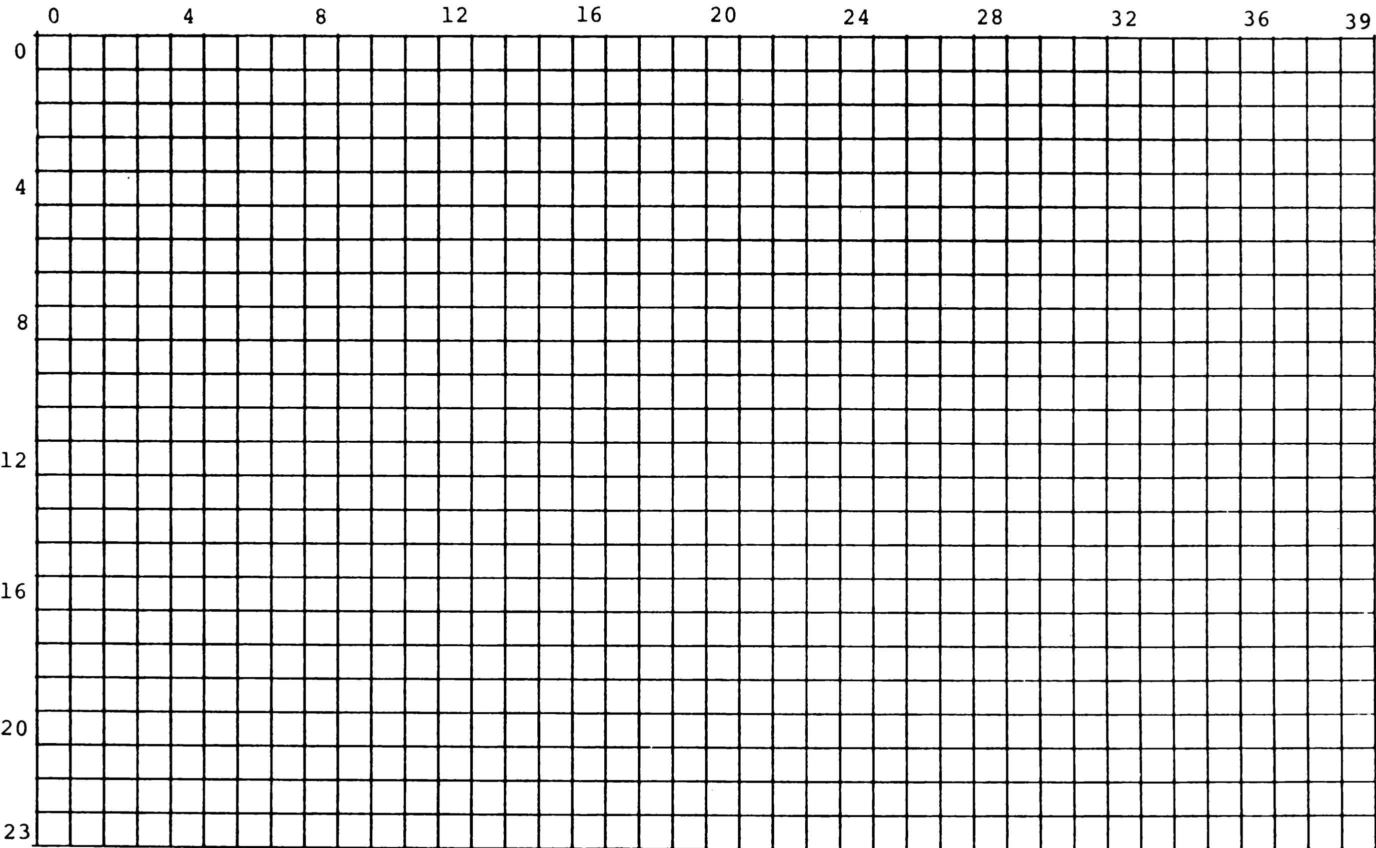


Modo Grafico 8

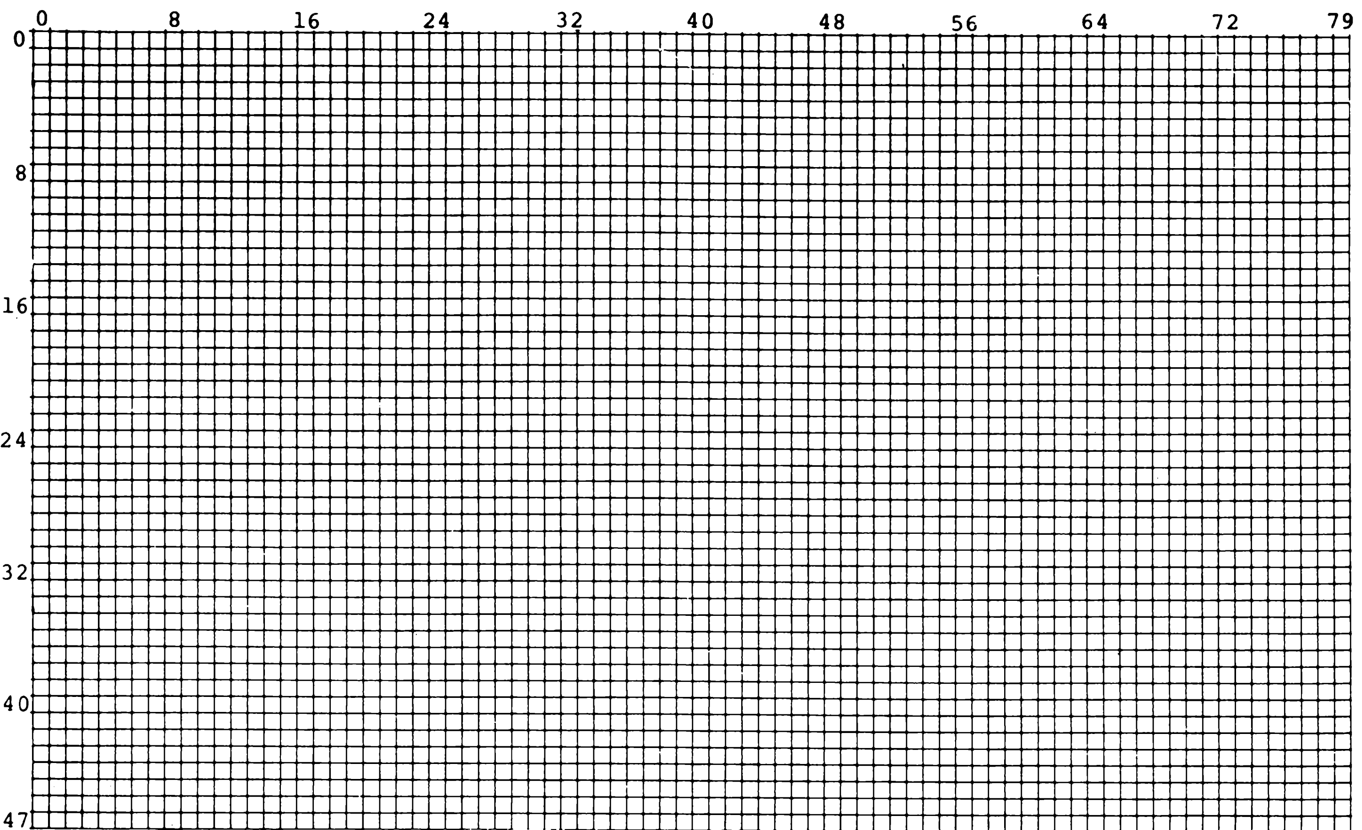
con Finestra Testo



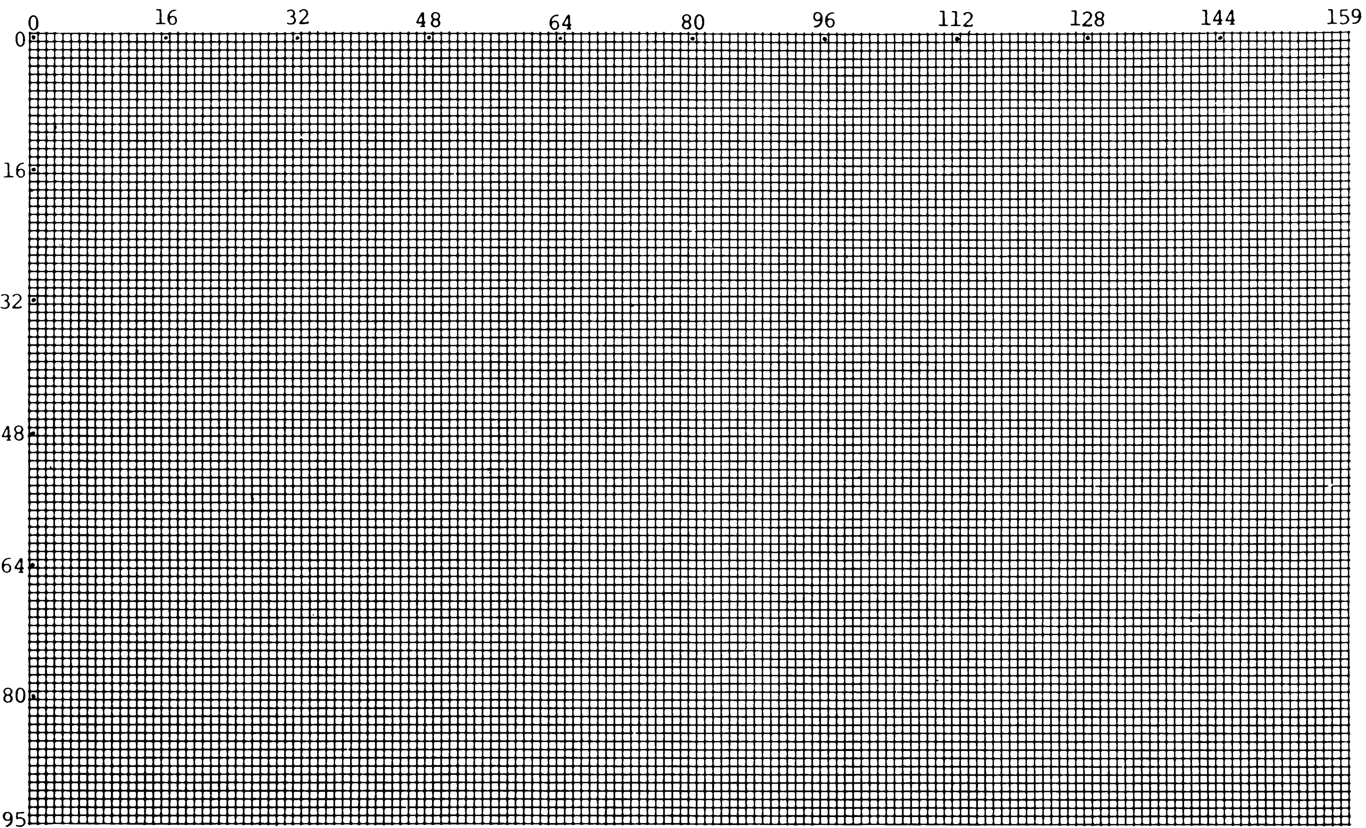
Modo Grafico 3



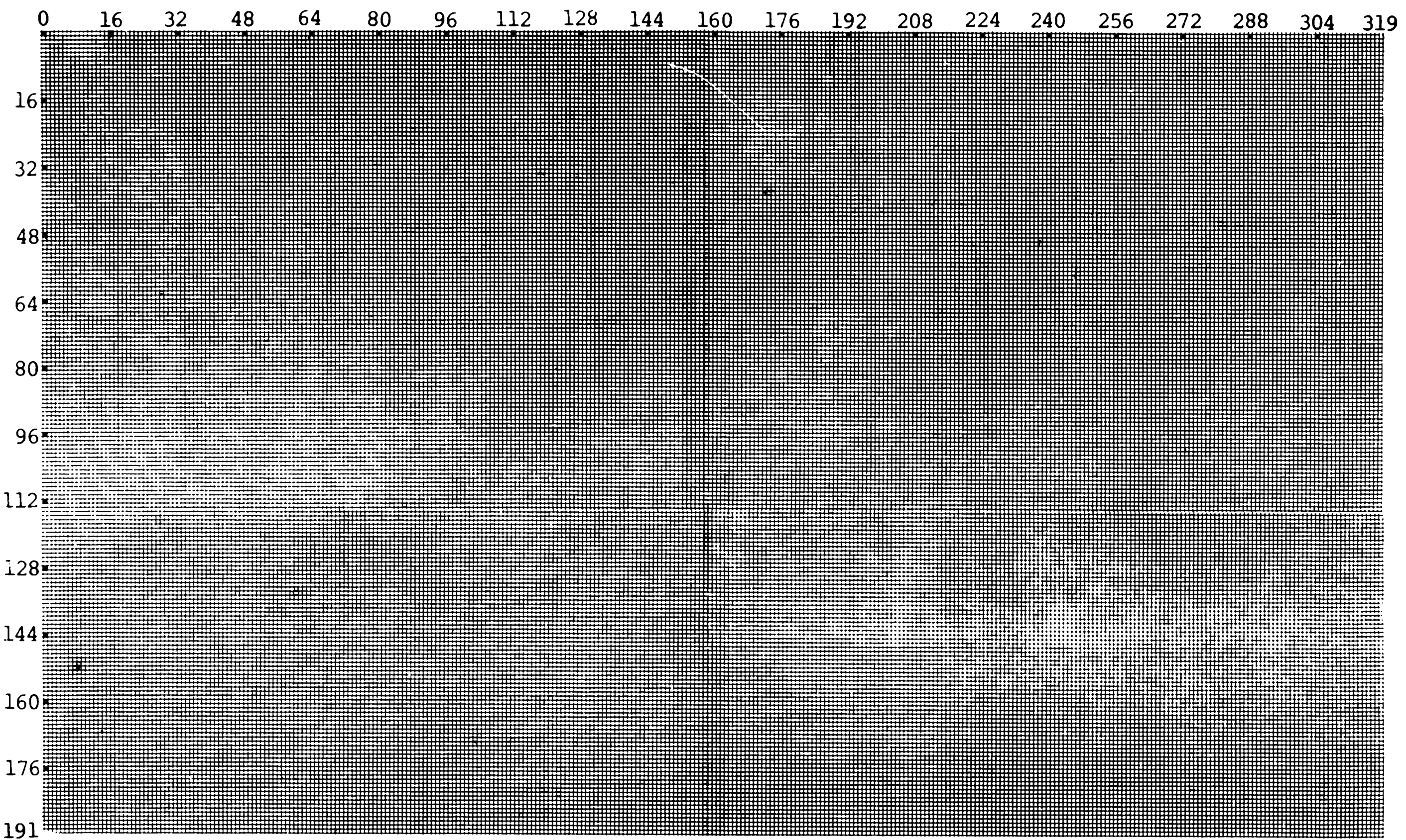
Modo Grafico 4 o 5



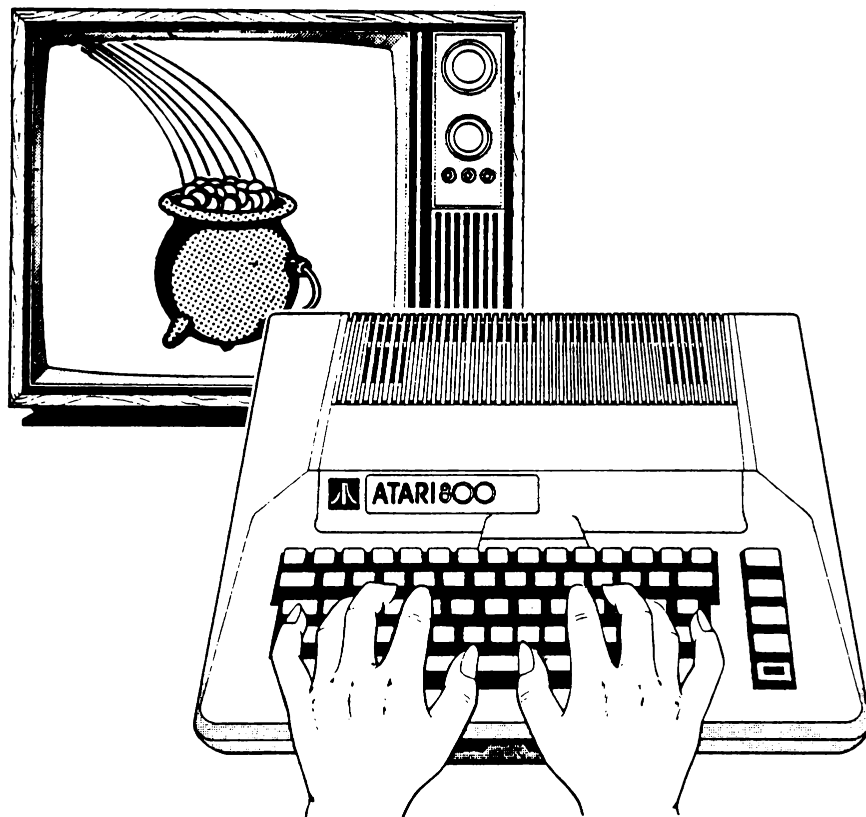
Modo Grafico 6 o 7



Modo Grafico 8.



Il colore ATARI



I Computer ATARI 400 e 800 posseggono superbe capacità cromatiche che vi consentono di disegnare oggetti ed evidenziare informazioni scegliendo tra 128 colori diversi. Il colore ATARI è abbastanza facile da usare, una volta accettato il fatto che non è del tutto facile da usare. In altri termini, non solo dovrete padroneggiare alcune regole generali strambe, ma probabilmente scoprirete che è più comodo lavorare col colore avendo sempre sotto mano questo libro.

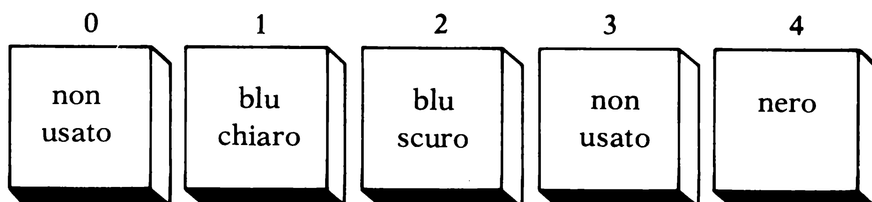
SETCOLOR

Per una conoscenza più approfondita del colore ATARI, dovete acquisire familiarità con due istruzioni: SETCOLOR e COLOR. Poiché soltanto la prima (abbreviata SE.) viene utilizzata in tutti i modi, grafici e testo, l'esamineremo per prima e ci dimenticheremo per un momento della COLOR.

Impiego di SETCOLOR in GR.0

Il computer ha dei registri del colore per memorizzare i colori; quali essi siano dipende dal modo in cui vi trovate.

Abitualmente nei registri del colore sono memorizzati da 3 a 5 colori. Per esempio, appena accendete il computer entrate automaticamente nel Modo 0 e i registri del colore sono caricati come segue...



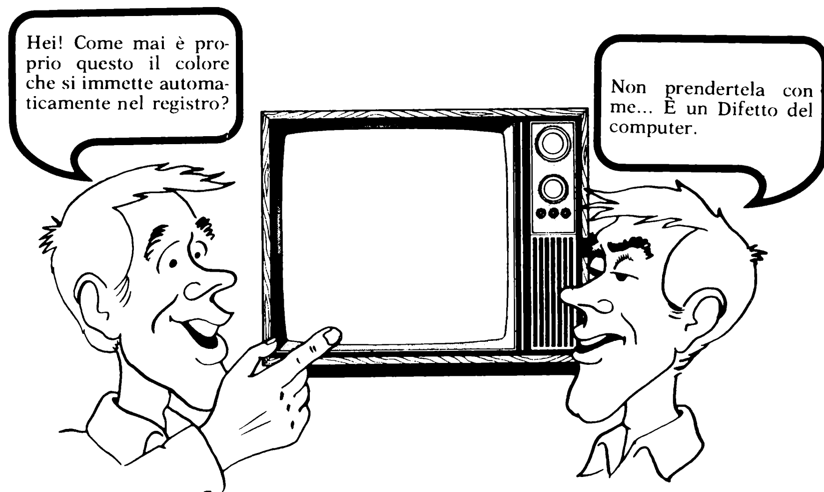
- il blu chiaro è collocato nel Registro 1, che comanda la luminosità dei caratteri, come per esempio quella della parola "READY" nell'angolo in alto a sinistra.
- il blu scuro è messo nel Registro 2, che impone il colore dello schermo o *"colore di fondo" (background).*
- il nero è posto nel Registro 4, che comanda il colore del *"bordo di contorno"* allo sfondo.
- nei registri 0 e 3 non è specificato alcun colore, poichè nel modo presente detti registri non sono impiegati.

I colori che vengono posti automaticamente nei registri del colore sono detti colori *di "default"* (*trad. lett.: per difetto; liberam.: predefiniti*).

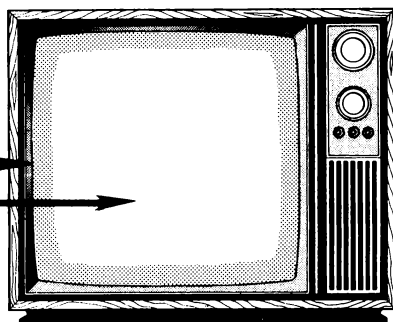
I colori di "default"

Tutte le volte che accendete il computer o cambiate modo grafico, l'elaboratore seleziona dei colori specifici da inserire nei registri del colore.

A voi la scelta se utilizzare tali colori predefiniti o definirne di nuovi attraverso la SETCOLOR.



NOTA: Mentre tutti i modi grafici hanno solo un colore di sfondo, i modi 0 e 8 hanno anche un bordo di contorno allo schermo, il cui colore di *default* è sempre il nero: ma imparerete presto come cambiarlo.



Come modificare i Colori

Avete la possibilità di modificare il colore di *background* o del bordo dello schermo in uno tra 16 colori base; inoltre, potete decidere quanto scuro o luminoso volete risulti il colore. Sedici colori moltiplicati per otto livelli di luminosità danno un totale di 128 colori tra cui scegliere.

I 16 colori di base tra cui scegliere sono

COLORI

0 = grigio
1 = arancio chiaro
2 = arancio
3 = arancio rosso
4 = rosa
5 = viola
6 = blu viola
7 = blu
8 = blu
9 = blu chiaro
10 = turchese
11 = blu verde
12 = verde
13 = verde giallo
14 = verde arancio
15 = arancio chiaro

Potete usare questo programma per selezionare sullo schermo tali colori

```
10 REM SCELTA COLORI
20 PRINT "↖":REM PULIZIA SCHERMO
30 POKE 752,1:REM DISABILITA CURSORE
40 POSITION 3,6:PRINT "DIGITA UN NUMERO
SEGUITO DA RETURN"
50 POSITION 6,9:PRINT "NUMERO COLORE";:INPUT
KOLOR
60. SETCOLOR 1,X,14:REM RENDE LUMINOSO IL TESTO
70 POSITION 9,11:PRINT "IL NUMERO DEL COLORE
E´ ";KOLOR
80 SETCOLOR 2,KOLOR,4
90 FOR RIT=1 TO 1000:NEXT RIT
100 GOTO 20
```

NOTA: Per evitare di frastornare il computer, è buona norma evitare l'utilizzo di nomi di variabili contenenti *parole chiave* del Basic, come COLOR o NEW. Si veda l'impiego della variabile KOLOR a linea 50.

Se i vostri colori non corrispondono perfettamente ai numeri, è possibile che il televisore o il monitor siano sregolati o che abbiano ON anziché OFF il comando di sintonia automatica (AFT). Se non riuscite ad avvicinarvi a un risultato accettabile, ogni Centro di Assistenza ATARI sarà in grado di controllare facilmente e presto l'uscita del vostro computer.

Oltre a saper comandare il colore dello schermo e del bordo, potrete anche controllare l'intensità dei caratteri sul video. Le seguenti informazioni vi metteranno in grado di comandare il colore dello sfondo, il colore del bordo e la luminosità dei caratteri...

Il primo numero che segue la SETCOLOR seleziona il *registro del colore* (1, 2 o 4 in GR.0). Per esempio, il colore di background è pilotato da una SETCOLOR 2.

SE. 2,4,8 cambierà lo schermo in un rosa leggermente vistoso, che è il colore 4 con luminosità 8.

Il numero successivo pilota il *colore corrente:*

SE. 2, da 0 a 15, 8

Qui può essere messo uno dei sedici colori precedentemente elencati.

Per esempio, SE. 2,6,8 cambierà lo schermo al color blu viola.

Il terzo numero controlla l'*intensità luminosa* del colore selezionato:

SE. 2,6, da 0 a 14

0 = INTENSITÀ MINIMA
14 = INTENSITÀ MASSIMA
(sono efficaci solo i numeri pari, così che 3 ha lo stesso effetto di 2, e 11 di 10).

Selezioniamo ora un altro registro di colore.

SE. 1, numero qualsiasi 0—14





Il numero è indifferente poiché in GR.0. il colore dei caratteri coincide sempre col colore di fondo.

0 = intensità minima
14 = intensità massima
Solo i numeri pari.

SE. 2,6,8
SE. 1,0,8

ATTENZIONE... se questi due ultimi numeri sono uguali, i caratteri scompaiono poiché lettere e sfondo assumono stesso colore e stessa intensità.

Ecco un programma che illustra come potete sfruttare questo fatto per far apparire e scomparire le parole.

```
10 PRINT "↖":REM PULIZIA SCHERMO
20 POKE 752,1:REM DISABILITA CURSORE
30 POSITION 12,11
40 PRINT "ADESSO MI VEDI..."
50 SETCOLOR 2,15,12
60 SETCOLOR 1,X,2
70 FOR RIT=1 TO 600:NEXT RIT
80 SETCOLOR 1,X,12:REM I CARATTERI DA ADESSO HANNO LA STESSA INTENSITA´ DELLO SF
ONDO
90 FOR RIT=1 TO 600:NEXT RIT
100 GOTO 50
```

Adesso cancellate il programma e digitate quest'altro che fa variare lentamente la luminosità delle lettere e dà l'illusione del lampeggio.

```
10 PRINT "↖":REM PULIZIA SCHERMO
20 POKE 752,1:REM DISABILITA CURSORE
30 POSITION 10,11
40 PRINT "ADESSO IL LAMPEGGIO..."
50 SETCOLOR 2,13,8:REM SFONDO
60 FOR J=1 TO 100
70 FOR FLASH=0 TO 14 STEP 2
80 SETCOLOR 1,X,FLASH
90 FOR RIT=1 TO 15:NEXT RIT
100 NEXT FLASH
110 NEXT J
```

Ricordate, SETCOLOR 4, –, – comanda il colore del bordo.



85 SETCOLOR 4,6,6:REM BORDER COLOR

Utilizzo di Setcolor nei Modi Grafici 1 e 2

Ora che dominate la tecnica di comando del colore nel modo 0, procediamo verso i Modi Grafici 1 e 2. Ma non dovete fare confusione!

GR. 1 e GR. 2 non hanno bordo, ma hanno la finestra testo. In entrambi i modi, potete comandare...

– il colore dello sfondo;
– il colore delle *PAROLE MAGGIORATE,* scritte dopo una PRINT #6; in lettere *maiuscole normali;*
– il colore delle *PAROLE MAGGIORATE,* scritte dopo una PRINT#6; in lettere *maiuscole in inverse video;*
– il colore delle *PAROLE MAGGIORATE,* scritte dopo una PRINT#6; in lettere *minuscole in inverse video;*
– il colore delle *PAROLE MAGGIORATE,* scritte dopo una PRINT#6; in lettere *minuscole normali;*
– il colore della finestra testo;
– la brillantezza del testo della finestra.

NOTA: Non tutte queste caratteristiche sono comandabili in maniera indipendente, ma dovreste arrivare ad ottenere qualsiasi effetto desiderato con pochi tentativi.



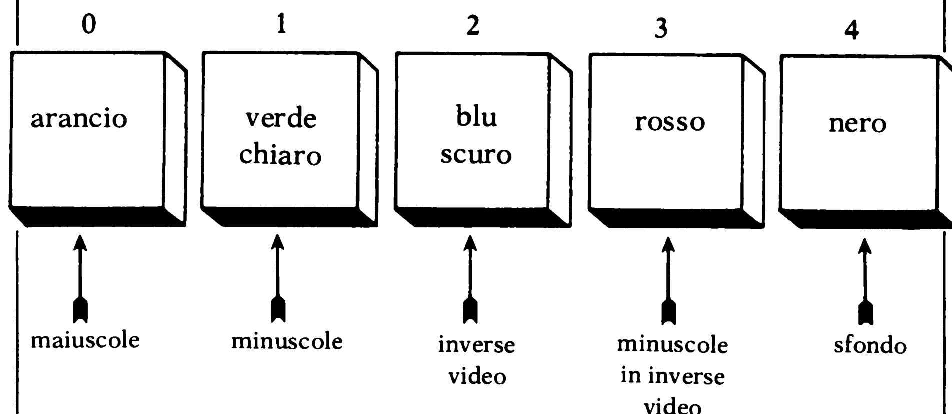
Colori predefiniti in GR.1 e GR.2

Poiché i registri del colore sono gli stessi sia in modo grafico 1 che in modo grafico 2, gli esempi seguenti verranno presentati in GR.2, con l'unica differenza che le parole risulteranno più grosse che in GR. 1.

Premete SYSTEM RESET
Quindi digitate...

GR. 2 RETURN

Fatto questo, il computer ha posto nei registri del colore i suoi colori di default per i modi grafici 1 e 2. E precisamente...



Per raggiungere i registri da 0 a 3, digitate PRINT#6; ed avrete a disposizione gli stili sopra illustrati; per accedere al registro 4, utilizzate la SETCOLOR 4,...,...

Un veloce ripasso

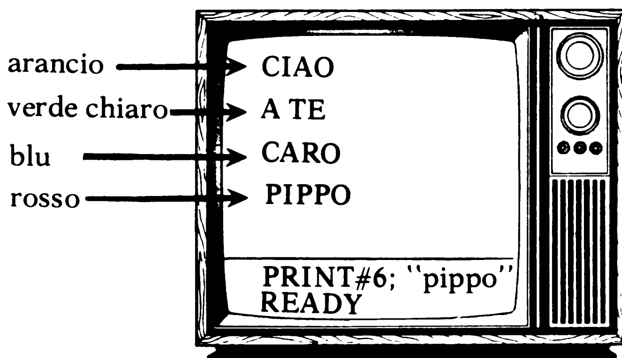
Per andare in INVERSE VIDEO	premete una volta [Atari]
Per tornare in normale	premetelo di nuovo
Per andare in minuscolo	premete una volta [CAPS LOWR]
Per tornare in normale	premete insieme [SHIFT] e [CAPS LOWR]

RICORDATE: il computer non accetta le istruzioni (come PRINT) in minuscolo.

Adesso provate...

```
5 GRAPHICS 2
10 PRINT #6;"CIAO"       scritto in MAIUSCOLO
20 PRINT #6;"a te"       scritto in minuscolo
30 PRINT #6;"CARO"       scritto in INVERSE VIDEO
40 PRINT #6;"pippo"      scritto in INVERSE VIDEO
                         minuscolo.
```

Dopo qualche tentativo lo schermo dovrebbe apparire così...



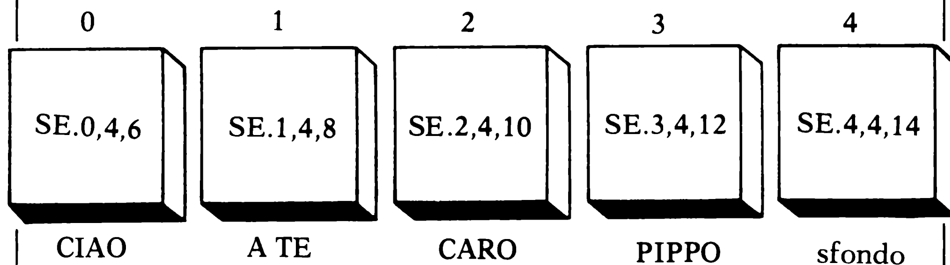
Se a questo punto siete *impantanati* senza speranza, riguardate il capitolo sul modo grafico 2 e ritornate all'inizio di questo capitolo. Entrare ed uscire dall'INVERSE VIDEO e dal minuscolo può essere complicato per il principiante; non fatevi scrupolo di premere [SYSTEM RESET] e digitare GR. 2 di nuovo, per ripartire da zero.

Proponiamoci ora di cambiare i colori dei registri con la SETCOLOR. Potremmo cominciare a cercare delle sfumature diverse di quel bel rosa vistoso che sembrava mi piacesse tanto.

SE. 0,4,6	colora CIAO quasi di ROSSO
SE. 1,4,8	colora A TE di ROSA
SE. 2,4,10	colora CARO e la finestra testo di un colore che andrebbe bene solo per il collare di un barboncino
SE. 3,4,12	colora PIPPO di un ROSA brillante.

Infine, diamo una schiarita anche allo sfondo...

SE. 4,4,14

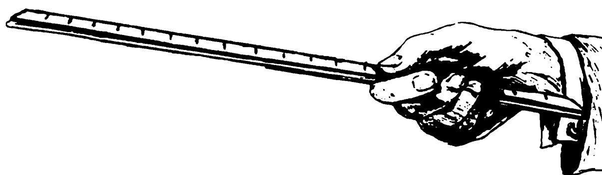


Il colore nei Modi GRAFICI da 3 a 8

Adesso che avete un'idea di come la SETCOLOR lavora nei modi testo, potete imparare ad usarla insieme con l'istruzione COLOR, utilizzata nei MODI GRAFICI da 3 a 8.

Ecco una regola generale.

I COLORI PER I MODI GRAFICI 3, 5 E 7 SONO GLI STESSI.



Quando entrate nei modi grafici 3, 5 o 7, i registri della SETCOLOR sono riempiti con i seguenti colori di default

0	1	2	3	4
arancio	verde chiaro	blu scuro	non usato	nero
				sfondo

Adesso potete fare una delle due cose:

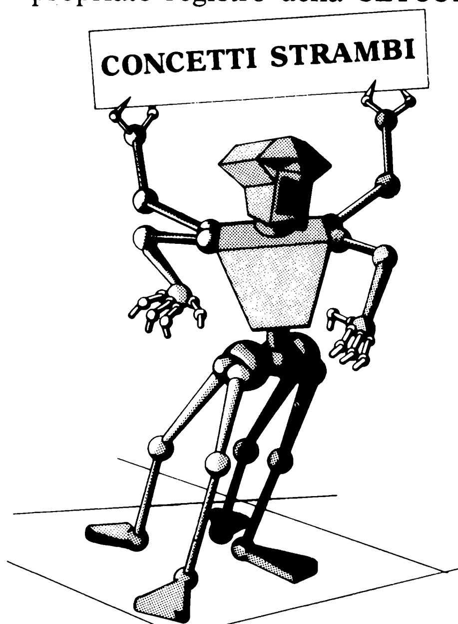
1. Modificare con la SETCOLOR uno dei colori di questi registri

— oppure —

2. Utilizzare l'istruzione COLOR per selezionare i registri 0, 1, 2 o 4, determinando così il colore dei punti o delle linee che disegnerete sullo schermo.

Impiego dell'istruzione COLOR

La COLOR è utilizzata nei modi propriamente GRAFICI (da 3 a 8), al fine di scegliere il registro di colore voluto. Quando desiderate selezionare o modificare il colore del prossimo punto o linea da tracciare, adoperate la COLOR con un numero che individui l'appropriato registro della SETCOLOR.



Voi penserete che volendo usare il colore del registro 0 si debba digitare COLOR 0; e parimenti COLOR 1, per il colore del registro 1. Eh, sarebbe troppo bello! Per costringere una buona dose di potenza dentro la cartuccia del vostro BASIC, i progettisti hanno deciso che voi avreste dovuto fare un piccolo sforzo di interpretazione. Succede quindi che le cose siano un pochino più complicate di come le avevate immaginate.

Eccovi un esempio che utilizza i colori di default nei registri 0, 1, 2 e 4. Digitate quanto segue, senza dimenticare di premere RETURN al termine di ogni linea.

```
10 GRAPHICS 3
20 COLOR 1
30 PLOT 2,2:DRAWTO 38,2
40 COLOR 2
50 PLOT 2,4:DRAWTO 38,4
60 COLOR 3
70 PLOT 2,6:DRAWTO 38,6
```

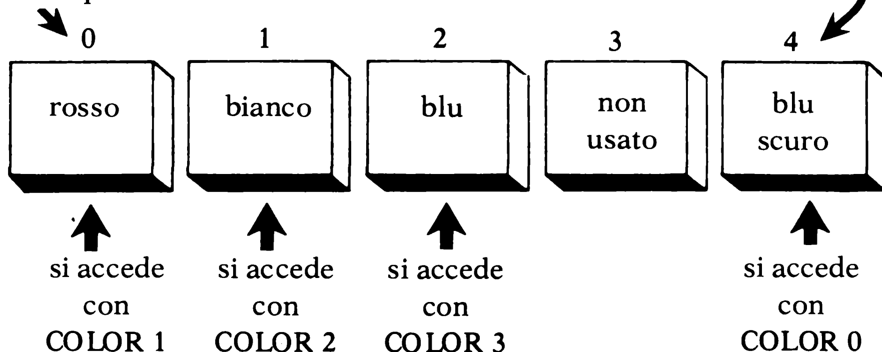
Osservate che per selezionare i registri 0, 1 e 2 si è impiegato COLOR 1, 2 e 3!

Inoltre, l'uso di SETCOLOR per cambiare i colori di quei registri provoca il cambiamento di colore delle linee che avete appena tracciato.

Osservate i risultati digitando quanto segue...

SE. 0,4,4	colora di ROSSO la prima linea.
SE. 1,0,14	colora di BIANCO la seconda linea e provoca anche un aumento della luminosità dei caratteri nella finestra testo, poiché la luminosità di questo registro pilota anche la brillantezza dei caratteri della finestra testo.
SE. 2,8,0	modifica il colore della terza linea e quello della finestra testo.
SE. 3,8,12	NON PRODUCE EFFETTI! Questo registro non è utilizzato in questo modo grafico.
SE. 4,7,4	colora lo sfondo con una diversa tonalità di blu

Adesso modificate il colore dei registri usando la SETCOLOR con questi numeri.

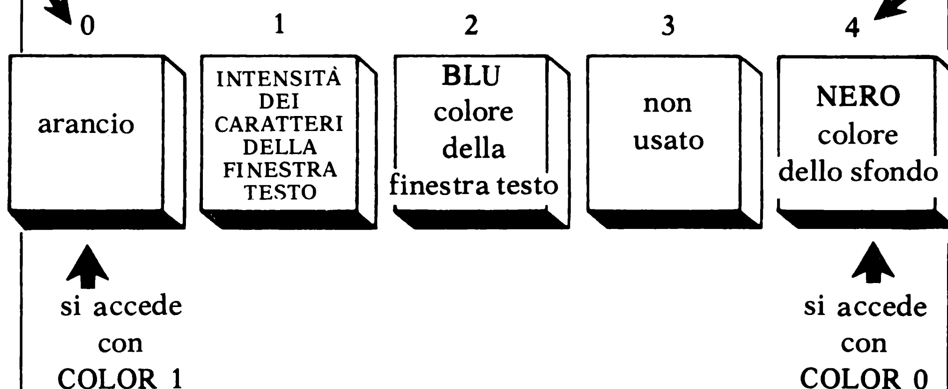


Tutto qui, vero?

Sbagliato! Vedete, il modo di operare della COLOR negli altri modi è leggermente differente.

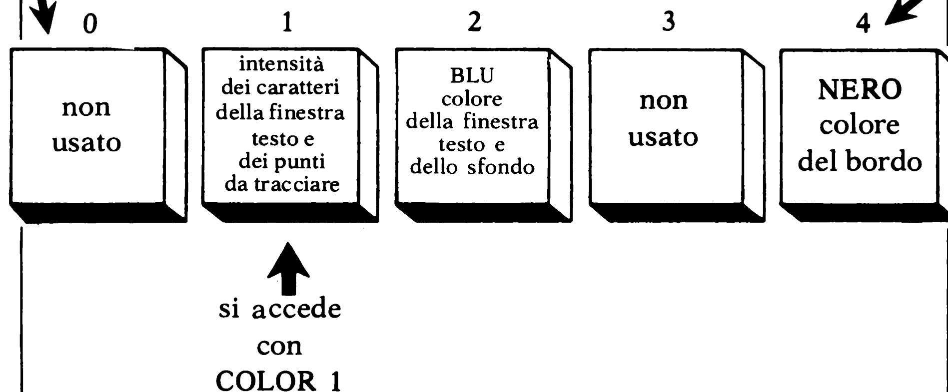
Modi GRAFICI 4 e 6

Per i modi grafici 4 e 6 sono utilizzabili solo due istruzioni COLOR. Con la SETCOLOR potrete modificare il colore di tutti i registri usando i numeri seguenti.



Modo GRAFICO 8

Nel modo grafico 8 con la SETCOLOR potrete modificare il colore di tutti i registri usando i seguenti numeri.



Riepilogo della SETCOLOR e della COLOR

I registri della SETCOLOR possono essere impiegati per una quantità di cose:

– il colore del bordo dello schermo;

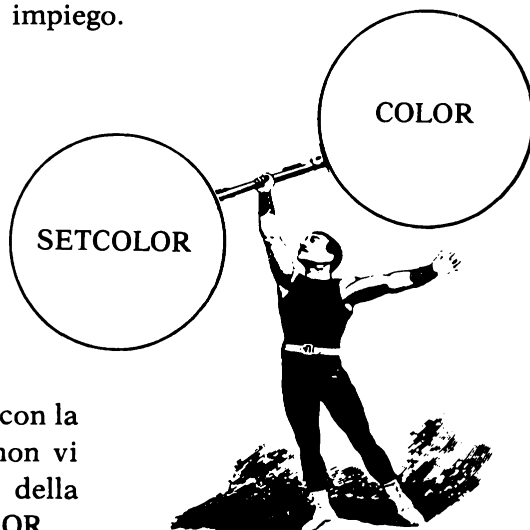
– il colore dei punti e delle linee disegnate;

– la luminosità dei caratteri sullo schermo;

– il colore degli oggetti disegnati con tecniche grafiche avanzate.

L'istruzione COLOR 1, 2, 3 o 4 è utilizzata (a partire dal modo 3) per notificare al computer che volete tracciare una linea avente il colore memorizzato in un particolare registro. Dovete esser certi che il numero di COLORe prescelto si accordi col registro della SETCOLOR impiegato, nel modo grafico in cui vi trovate.

Se non avete le idee chiare, non preoccupatevi; tutto viene ancora ripetuto nel capitolo *"Legame tra la SETCOLOR e la POKE dei colori"*; al termine della lettura del capitolo, troverete una scheda riepilogativa, di comodo impiego.



Con un piccolo ripasso e con la pratica, scoprirete che non vi sono difficoltà nell'uso della SETCOLOR e della COLOR.



Il suono



Provate questo programma...

```
10 FOR Z=0 TO 255
20 SOUND 0,Z,10,10
30 NEXT Z
```

Accertatevi che il volume del vostro televisore sia acceso e mandate in esecuzione il programma.

Questo è un loop elementare di comando del suono, che su molti altri computer richiederebbe un monumentale lavoro di programmazione.

Adesso aggiungete...

```
40 FOR Z=255 TO 0 STEP -1
50 SOUND 0,Z,10,10
60 NEXT Z
70 GOTO 10
```

e mandate in esecuzione questo nuovo programma.

Fondamenti dell'istruzione SOUND

Vi sono modi anche assai complicati per emettere suoni con gli Home Computer ATARI, ma gli elementi essenziali per creare il suono in Basic possono riassumersi come segue:

1. Per produrre suoni, dovete dire al computer che intendete farlo.
2. Dovete specificare un numero da 0 a 3, per utilizzare un registro del suono (ve ne sono 4 in tutto).
3. Definite quindi l'altezza del suono, tra 0 e 255.
4. Ancora, definite un livello di distorsione tra 0 e 14 (solo i numeri pari). Il valore 10 è per un tono puro, senza distorsione.
5. Infine, fissate il volume (tutti sanno cosa vuol dire). I valori per il volume vanno da 0 a 15.

```
SOUND 3,45,10,10
```

qui vanno le virgole

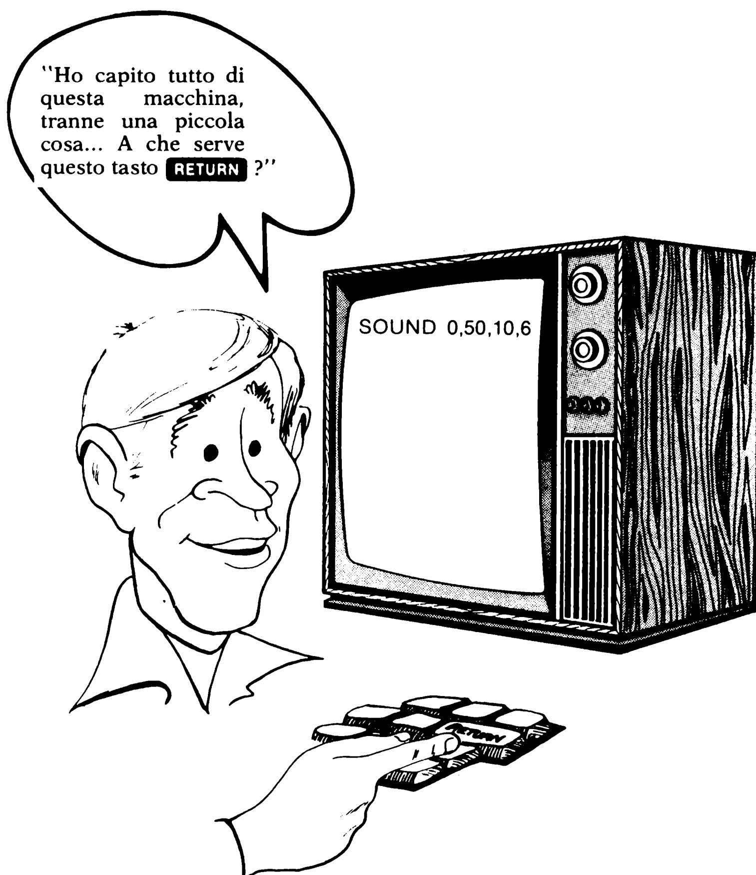
Osservate che potete produrre un suono totalmente differente con ognuno dei quattro registri del suono.

PREMETE **SYSTEM RESET**

Ora provate qualche armonia in modo diretto. Digitate...

```
SOUND 0,50,10,6
SOUND 1,100,10,6
SOUND 2,150,10,6
SOUND 3,200,10,6
```

Ricordatevi di premete il tasto **RETURN** ed alzare il volume del televisore.



Per *"spegnere"* il suono, usate SOUND 0,0,0,0

Fate attenzione a indicare il numero giusto del registro del suono da spegnere.

```
10 REM ORGANO DA TASTIERA
20 PRINT "ALZA IL VOLUME DEL TELEVISORE E PREMI UN TASTO QUALSIASI, ESCLUSO BREA
K"
30 OPEN #1,4,0,"K:"
40 GET #1,X
50 SOUND 0,X,10,8:POKE 712,X
60 FOR RIT=0 TO 75:NEXT RIT
70 SOUND 0,0,0,0
80 GOTO 40
```

Queste istruzioni recuperano il codice ASCII del tasto che avete premuto. La cosa è un tantino avanzata e comunque non effettivamente importante per il suono. Ulteriori informazioni sull'uso della codifica ASCII si possono trovare nel capitolo sulle Funzioni.

Provate questo programma così com'è scritto; poi cancellate l'istruzione SOUND 0,0,0,0 ed osservate la differenza.

Eccovi un programma da eseguire insieme con *"Stormy Weather"*...

```
10 GRAPHICS 7+16:SETCOLOR 4,0,0:SETCOLOR 1,0,4
20 COLOR 2
30 PLOT 2,2:DRAWTO 30,34:DRAWTO 78,40:DRAWTO 100,57:DRAWTO 110,50:DRAWTO 140,76
40 FOR LEV=1 TO 255
50 SOUND 0,LEV,8,10
60 IF LEV=8 THEN SETCOLOR 1,0,14
70 NEXT LEV
80 SETCOLOR 1,0,0
90 FOR RIT=1 TO 200:NEXT RIT
100 GOTO 10
```

Troverete altri esempi relativi al suono nel capitolo sui controllori di paddle.

POKE-ando qua e là

All'accensione del vostro computer ATARI, alcune locazioni ben definite di memoria sono riempite automaticamente con valori che influenzano tutto il comportamento successivo.

Per esempio, nel modo grafico 0 standard, il numero posto nella locazione 710 pilota il colore dello schermo: all'accensione del computer, in questa locazione viene posto 148. Si scopre che 148 corrisponde al colore BLU che già vi è familiare per lo schermo; sapete anche come fare per modificarlo con la SETCOLOR, ma esiste un altro modo...

Se digitate...

POKE 710,79

... osserverete che lo schermo si è tinto di un colore piuttosto squallido.

Adesso digitate...

POKE 712,79

... e scoprirete che anche il bordo ha assunto lo stesso colore.

Da ciò, potete concludere che nel modo grafico 0 il colore dello schermo è comandato dalla locazione 710; e il colore del bordo, dalla locazione 712. Il numero 79 è semplicemente il numero associato ad un colore particolare.

Provate a digitare il programma che segue...

```
10 FOR X=0 TO 254 STEP 2
20 POKE 710,X
30 NEXT X
```

Eseguitelo...

Adesso provate...

```
10 PRINT " "
20 FOR X=0 TO 254 STEP 2
30 PRINT X
40 POKE 710,X
50 FOR R=1 TO 50:NEXT R
60 NEXT X
```

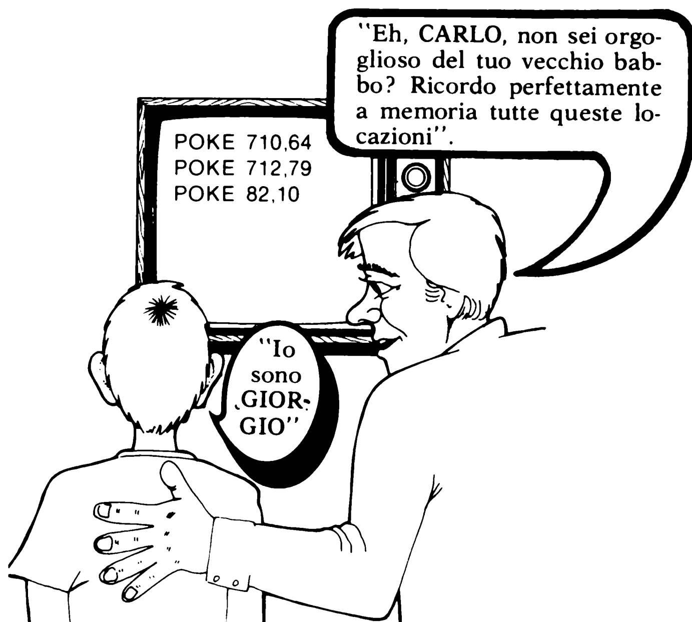
Se volete arrestare l'esecuzione e verificare una combinazione particolare di numero e colore, premete simultaneamente CTRL e 1 Premeteli di nuovo per ripartire.

Adesso, aggiungete ancora una linea al programma, facendo scomparire il cursore.

```
5 POKE 752,1
```

Per farlo ricomparire, basta depositare uno 0 nella locazione 752.

In certi casi, per far scomparire il cursore è necessaria la POKE 755,1; se 752,1 non funziona, provate con 755,1.



PEEK e POKE

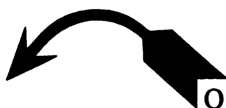
È possibile anche *spiare* una locazione di memoria solo per vedere il valore in essa contenuto, senza variarlo. Ciò si ottiene con l'istruzione PEEK.

Per esempio...

premete SYSTEM RESET

quindi digitate...

PRINT PEEK(710)



Osservate l'impiego delle parentesi intorno al numero della locazione che volete sondare.

Dovreste ricavarne 148, il numero associato allo schermo BLU standard.

Ora provate ad eseguire una PEEK della locazione 82, che comanda il *margine sinistro* del computer.

PRINT PEEK(82)

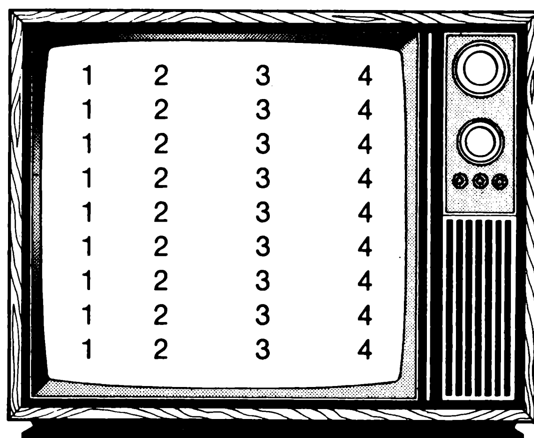
Dovreste ottenere 2, pari al numero di spazi che abitualmente il computer lascia come margine sinistro.

Torniamo alle POKE

Provate questo programma, che assegna lo stesso valore ai margini sinistro e destro.

```
10 POKE 82,19:POKE 83,19
20 PRINT "CIAO A TUTTI"
30 FOR RIT=1 TO 350:NEXT RIT:GOTO 20
```

Disponete adesso dei nuovi margini, fino a che non li riportate ai valori normali, oppure premete SYSTEM RESET, o non spegnete il computer.



Nuova ampiezza di tabulazione con la POKE

Quando stampate parole o numeri separati da virgole, il computer predispone normalmente *10 spazi* per ogni colonna. Se volete cambiare tale valore, depositate nella locazione 201 un nuovo valore di ampiezza di tabulazione.

```
10 POKE 201,2:POKE 82,0
20 PRINT :PRINT A,B,C,D,E,F,G
30 A=A+1:B=B+1:C=C+1:D=D+1:E=E+1:F=F+1:G=G+1
40 GOTO 20
```

Ed ecco un programma che combina le POKE di registri del colore con le POKE di nuovi margini.

```
10 POKE 710,0:POKE 752,1
20 FOR Z=1 TO 38:POKE 82,Z
30 PRINT "\"
40 POKE 712,Z*5
50 SOUND 0,Z*8,10,10
60 NEXT Z
70 FOR X=37 TO 1 STEP -1
80 POKE 82,X
90 POKE 712,X*5
100 PRINT "/"
110 SOUND 0,X*8,10,10
120 NEXT X
130 RUN
```

La locazione 755 ha qualche effetto interessante sui testi in modo normale e modo INVERSE VIDEO. Il valore normale per questa locazione è 2, così che il ritorno alla situazione originale si ottiene con POKE 755,2.

Altri valori nella locazione 755 producono gli effetti seguenti...

POKE 755,0 – fa passare da INVERSE VIDEO al modo normale e disattiva il cursore.

POKE 755,1 – fa scomparire ogni cosa in INVERSE VIDEO e disattiva il cursore.

POKE 755,2 – riporta tutto alla normalità.

POKE 755,3 – produce blocchi pieni, tranne ciò che è in INVERSE VIDEO.

POKE 755,4 – scrive i caratteri a testa in giù col cursore spento.

POKE 755,6 – scrive i caratteri a testa in giù col cursore ancora attivo.

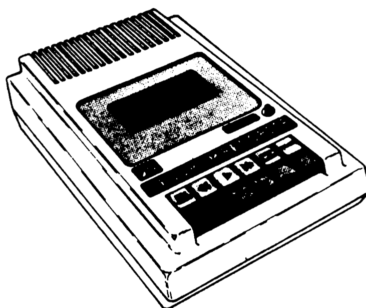
Il programma che segue mostra la stessa parola in INVERSE VIDEO, modificata dall'impiego di diversi valori per la locazione 755.

```
10 PRINT "↖":REM PULIZIA SCHERMO
20 POKE 752,1:REM DISABILITA CURSORE
30 POSITION 16,12:PRINT "SORPRESA":REM DIGITARE "SORPRESA" IN INVERSE VIDEO
40 FOR Z=0 TO 4
50 POKE 755,Z
60 FOR RIT=1 TO 400:NEXT RIT
70 NEXT Z
80 GOTO 40
```

Ed ecco qualche altra locazione interessante per la POKE.

POKE 54018,52 – fa partire il Registratore di Programmi ATARI 410 (Il tasto PLAY del registratore deve essere abbassato).

POKE 54018,60 – arresta il Registratore.



Con una POKE 54018,52 potete riprodurre una normale musicassetta attraverso l'altoparlante del vostro televisore. Prima di cominciare, assicuratevi che sul Registratore 410 il nastro sia riavvolto ed il tasto PLAY sia abbassato.

Eccovi un breve programma che mostra come si possa utilizzare il computer per avviare e fermare la musica.

```
10 PRINT "↖":POKE 752,1:DIM RI$(1),ST$(1)
20 PRINT "PREMI -S- E -RETURN- PER ASCOLTARE LA MUSICA"
30 INPUT RI$
40 IF RI$="S" THEN POKE 54018,52:GOTO 60
50 GOTO 20
60 PRINT "↖":PRINT "PER ARRESTARE LA MUSICA PREMI -N- E -RETURN-"
70 INPUT ST$
80 IF ST$="N" THEN POKE 54018,60:GOTO 20
90 GOTO 60
```

Il computer può eseguire una PEEK alla locazione 53279 per vedere se fino a quel momento qualcuno ha premuto i tasti START, SELECT o OPTION.

Un 6 significa che è stato premuto START.
Un 5 significa che è stato premuto SELECT.
Un 3 significa che è stato premuto OPTION.

Eccovi il frammento d'apertura di un programma di giochi:

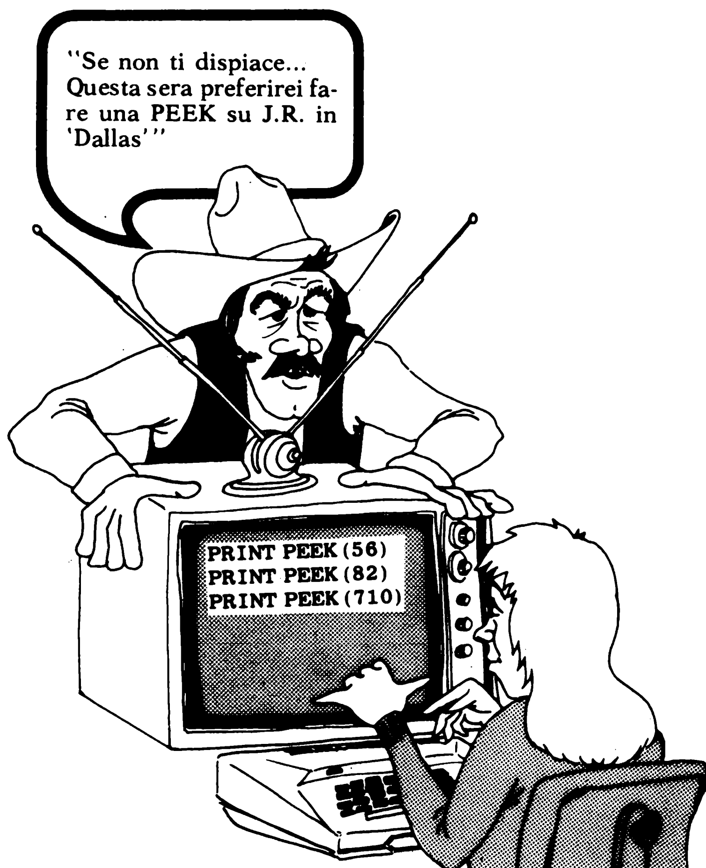
```
10 PRINT "TASTO -START- PER PARTIRE"
20 IF PEEK(53279)=6 THEN GOTO 40
30 GOTO 20
40 PRINT "PRONTO PEE COMINCIARE"
```

Nel computer vi sono anche locazioni che possono essere usate come *temporizzatori*.

Il cuore di questo programma contasecondi è alla linea 50. Dovreste utilizzarlo per sperimentare la realizzazione di orologi, timer interni, segnalatori di stop. Ricordate però che produrre un'immagine video elaborata richiede tempo al computer; così che dovrete mettere d'accordo le vostre esigenze di temporizzazione con la complessità delle immagini.

```
10 PRINT "↖":POKE 752,1
20 POSITION 5,11:PRINT "-START- PER PARTIRE COL CONTATORE":IF PEEK(53279)=6 THEN
 PRINT "↖":GOTO 40
30 GOTO 20
40 POKE 18,0:POKE 19,0:POKE 20,0
50 SEC=INT((PEEK(18)*65535+PEEK(19)*256+PEEK(20))/60)
60 POSITION 19,11:PRINT SEC
70 IF SEC=60 THEN PRINT "↖":GOTO 40
80 GOTO 50
```

Una lista più dettagliata delle locazioni in cui potete caricare valori con la POKE si trova nel capitolo *"memory locations"* del Manuale di Riferimento del Basic ATARI.



NOTA: È facilissimo per voi mandare in CRASH il programma (vale a dire farlo *"impiantare"* temporaneamente) introducendo numeri con la POKE in locazioni sconosciute di memoria. Se ciò si verifica, dovrete spegnere il computer e riaccenderlo. Alcuni raccomandano di non caricare valori in zone della memoria sconosciute, poiché potreste fortuitamente cancellare un dischetto o qualcosa del genere. Invece leggere da memoria con la PEEK non può mai provocare danno.

Legame tra la SETCOLOR e la POKE dei colori

Come avete appreso, i computer ATARI hanno 128 colori e al tempo stesso ne hanno 16. A loro modo, entrambe queste affermazioni sono valide: avete 128 varianti di colore entro cui scegliere.

Cominciate con i 16 colori di base...

0 = grigio
1 = oro
2 = arancio
3 = rosso-arancio
4 = rosa
5 = rosa-viola
6 = blu-viola
7 = blu
8 = altro blu
9 = blu chiaro
10 = turchese
11 = blu-verde
12 = verde
13 = verde-giallo
14 = verde-arancio
15 = arancio chiaro

...ognuno di questi colori ha 8 livelli di luminosità o brillantezza (dal più sicuro al più chiaro)

16 colori × 8 intensità = 128 colori tra cui scegliere!

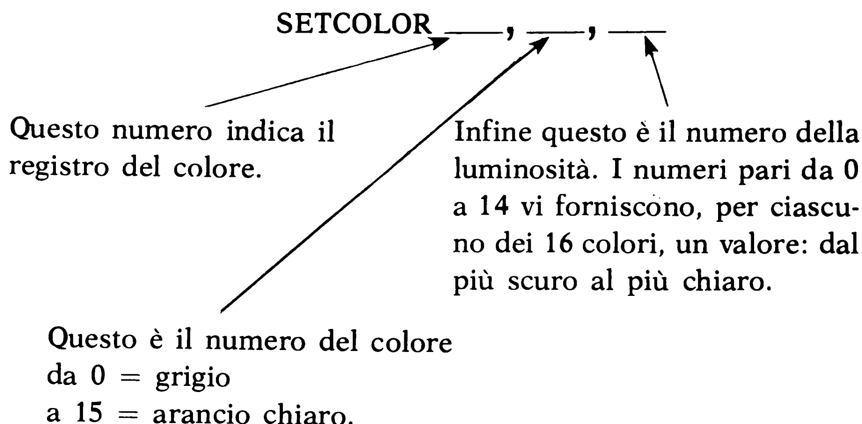
Per vedere tutti i 128 colori utilizzate il programmino:

```
10 PRINT "↖":POKE 752,1
20 FOR Z=0 TO 254 STEP 2
30 POSITION 19,12:PRINT Z
40 POKE 710,Z
50 FOR RIT=1 TO 25:NEXT RIT
60 NEXT Z
```

NOTA: si è utilizzato il loop da 0 a 254 (e non da 0 a 128) poichè influiscono sul colore dello schermo soltanto i numeri pari.

Il formato dell'istruzione SETCOLOR _, _, _ è stato previsto per favorire una miglior comprensione ed impiego della coppia colore/intensità, e talvolta risulta comodo; ma spesso appare più conveniente l'utilizzo delle POKE. La comprensione del legame tra i due modi di operare vi faciliterà l'esistenza.

Ecco un rapido riassunto del modo di operare della SETCOLOR. Quando utilizzate una SETCOLOR...



Definire i colori mediante le POKE significa semplicemente inserire i valori equivalenti di colore (compresi tra 0 e 254) nei rispettivi registri del colore.

Registri del colore	704	Valori di colore	0
fino a	..		2
	712	fino a	..
			252
			254

Per esempio, in modo grafico 0...

SETCOLOR 2, ____, ____

oppure

POKE 710, ____

comandano entrambe il colore di fondo dello schermo.

Provate gli esempi che seguono e consultate la *Tabella di Equivalenza* tra SETCOLOR e POKE per una comprensione più completa dei legami tra le due.

SE. 2,3,6
darà lo stesso colore della...
POKE 710,54

oppure

SE. 2,15,4
equivale a...
POKE 710,244



SE. 4,4,6

darà lo stesso colore della...

POKE 712,70

oppure

SE. 4,7,12

equivale a...

POKE 712,124

Tabella di Equivalenza tra SETCOLOR e POKE

SE. num-reg, ___, ___

colore= 0

		VALORE EQUIVALENTE PER LA POKE
grigio	LUMINANCE= 0	0
	LUMINANCE= 2	2
	LUMINANCE= 4	4
	LUMINANCE= 6	6
	LUMINANCE= 8	8
	LUMINANCE= 10	10
	LUMINANCE= 12	12
	LUMINANCE= 14	14

colore= 1

oro	LUMINANCE= 0	16
	LUMINANCE= 2	18
	LUMINANCE= 4	20
	LUMINANCE= 6	22
	LUMINANCE= 8	24
	LUMINANCE= 10	26
	LUMINANCE= 12	28
	LUMINANCE= 14	30

colore= 2

arancio	LUMINANCE= 0	32
	LUMINANCE= 2	34
	LUMINANCE= 4	36
	LUMINANCE= 6	38
	LUMINANCE= 8	40
	LUMINANCE= 10	42
	LUMINANCE= 12	44
	LUMINANCE= 14	46

colore= 3

rosso-arancio	LUMINANCE= 0	48
	LUMINANCE= 2	50
	LUMINANCE= 4	52
	LUMINANCE= 6	54
	LUMINANCE= 8	56
	LUMINANCE= 10	58
	LUMINANCE= 12	60
	LUMINANCE= 14	62

colore= 4

		VALORE EQUIVALENTE PER LA POKE
rosa	LUMINANCE= 0	64
	LUMINANCE= 2	66
	LUMINANCE= 4	68
	LUMINANCE= 6	70
	LUMINANCE= 8	72
	LUMINANCE= 10	74
	LUMINANCE= 12	76
	LUMINANCE= 14	78

colore= 5

rosa-viola	LUMINANCE= 0	80
	LUMINANCE= 2	82
	LUMINANCE= 4	84
	LUMINANCE= 6	86
	LUMINANCE= 8	88
	LUMINANCE= 10	90
	LUMINANCE= 12	92
	LUMINANCE= 14	94

colore= 6

blu-viola	LUMINANCE= 0	96
	LUMINANCE= 2	98
	LUMINANCE= 4	100
	LUMINANCE= 6	102
	LUMINANCE= 8	104
	LUMINANCE= 10	106
	LUMINANCE= 12	108
	LUMINANCE= 14	110

colore= 7

blu	LUMINANCE= 0	112
	LUMINANCE= 2	114
	LUMINANCE= 4	116
	LUMINANCE= 6	118
	LUMINANCE= 8	120
	LUMINANCE= 10	122
	LUMINANCE= 12	124
	LUMINANCE= 14	126

colore= 8 — blu

		VALORE EQUIVALENTE PER LA POKE
	LUMINANCE= 0	128
	LUMINANCE= 2	130
	LUMINANCE= 4	132
blu	LUMINANCE= 6	134
	LUMINANCE= 8	136
	LUMINANCE= 10	138
	LUMINANCE= 12	140
	LUMINANCE= 14	142

colore= 9

	LUMINANCE= 0	144
	LUMINANCE= 2	146
	LUMINANCE= 4	148
	LUMINANCE= 6	150
blu chiaro	LUMINANCE= 8	152
	LUMINANCE= 10	154
	LUMINANCE= 12	156
	LUMINANCE= 14	158

colore= 10

	LUMINANCE= 0	160
	LUMINANCE= 2	162
	LUMINANCE= 4	164
turchese	LUMINANCE= 6	166
	LUMINANCE= 8	168
	LUMINANCE= 10	170
	LUMINANCE= 12	172
	LUMINANCE= 14	174

colore= 11

	LUMINANCE= 0	176
	LUMINANCE= 2	178
	LUMINANCE= 4	180
	LUMINANCE= 6	182
blu-verde	LUMINANCE= 8	184
	LUMINANCE= 10	186
	LUMINANCE= 12	188
	LUMINANCE= 14	190

colore= 12

		VALORE EQUIVALENTE PER LA POKE
	LUMINANCE= 0	192
	LUMINANCE= 2	194
	LUMINANCE= 4	196
verde	LUMINANCE= 6	198
	LUMINANCE= 8	200
	LUMINANCE= 10	202
	LUMINANCE= 12	204
	LUMINANCE= 14	206

colore= 13

	LUMINANCE= 0	208
	LUMINANCE= 2	210
	LUMINANCE= 4	212
verde-	LUMINANCE= 6	214
giallo	LUMINANCE= 8	216
	LUMINANCE= 10	218
	LUMINANCE= 12	220
	LUMINANCE= 14	222

colore= 14

	LUMINANCE= 0	224
	LUMINANCE= 2	226
	LUMINANCE= 4	228
verde-	LUMINANCE= 6	230
arancio	LUMINANCE= 8	232
	LUMINANCE= 10	234
	LUMINANCE= 12	236
	LUMINANCE= 14	238

colore= 15

	LUMINANCE= 0	240
	LUMINANCE= 2	242
	LUMINANCE= 4	244
arancio	LUMINANCE= 6	246
chiaro	LUMINANCE= 8	248
	LUMINANCE= 10	250
	LUMINANCE= 12	252
	LUMINANCE= 14	254

Talvolta può essere facile e interessante da sperimentare l'introduzione dei numeri da 0 a 254 nei registri del colore 704 e 712. Troverete molti importanti risultati che impiegano le locazioni da 708 a 712; successivamente imparerete molto di più sulle locazioni da 704 a 707.

Esempio:

```
10 FOR Z=0 TO 254 STEP 2
20 POKE 710,Z
30 FOR RIT=1 TO 35:NEXT RIT
40 NEXT Z
```

produrrà lo stesso risultato di questa versione con la SETCOLOR...

```
10 FOR K=0 TO 15:REM LOOP SUI 16 COLORI
20 FOR L=0 TO 14 STEP 2:REM LOOP SUGLI 8 VALORI DI INTENSITA' DI OGNI COLORE
30 SETCOLOR 2,K,L
40 FOR RIT=1 TO 35:NEXT RIT
50 NEXT L
60 NEXT K
```

Tuttavia, immaginiamo che vogliate passare gradualmente da uno schermo rosso scuro a uno schermo rosso chiaro, più e più volte. Allo scopo, tutto quello che dovreste ricordare (o andare a cercare) è che il rosso è il numero 4 collocato in seconda posizione nella SETCOLOR; dovreste quindi passare attraverso tutti i valori di luminosità, da 0 a 14.

Questa forma è assai più semplice che non dover ricordare tutti i valori equivalenti per la POKE

```
10 FOR N=0 TO 14 STEP 2
20 SETCOLOR 2,4,N
30 FOR RIT=1 TO 35:NEXT RIT
40 NEXT N
50 GOTO 10
```

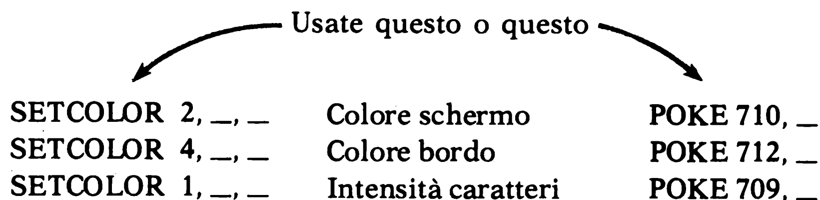
```
10 FOR N=64 TO 78 STEP 2
20 POKE 710,N
30 FOR RIT=1 TO 35:NEXT RIT
40 NEXT N
50 GOTO 10
```

Concludendo, talvolta l'approccio da usare passa per la SETCOLOR, altre volte la POKE è la via giusta per la *colorgrafica*. Appena imparerete nuovi modi grafici, scoprirete che servono tanto la SETCOLOR quanto la POKE; esercitarvi con entrambe vi condurrà sempre più innanzi sulla strada del colore.

Locazioni corrispondenti ai registri dello SETCOLOR

Nel seguito sono illustrate le equivalenze dei registri. Sta a voi decidere se usare una *SETCOLOR (registro, colore, intensità)* o una *POKE (da 0 a 254).*

Modo Grafico 0



SETCOLOR 2, _, _	Colore schermo	POKE 710, _
SETCOLOR 4, _, _	Colore bordo	POKE 712, _
SETCOLOR 1, _, _	Intensità caratteri	POKE 709, _

Modi Grafici 1 e 2

SETCOLOR 2, _, _	Finestra testo.	POKE 710, _
SETCOLOR 1, _, _	Intensità caratteri della finestra testo.	POKE 709, _
SETCOLOR 4, _, _	Colore di fondo.	POKE 712, _
SETCOLOR 0, _, _	Colore del testo maggiorato (PRINT #6;) introdotto in caratteri normali maiuscoli.	POKE 712, _
SETCOLOR 2, _, _	Colore del testo maggiorato (PRINT #6;) introdotto in INVERSE VIDEO. *Nota:* è lo stesso colore della finestra testo.	POKE 710, _
SETCOLOR 3, _, _	Colore del testo maggiorato (PRINT #6;) introdotto in caratteri minuscoli in INVERSE VIDEO.	POKE 711, _
SETCOLOR 1, _, _	Colore del testo maggiorato (PRINT #6;) introdotto in caratteri minuscoli. *Nota:* comanda anche l'intensità dei caratteri normali nella finestra testo e può farli scomparire se l'intensità di questi uguaglia quella dello sfondo.	POKE 709, _

Modi Grafici 3, 5 e 7

SETCOLOR 4, _, _	Colore di fondo.	POKE 712, _
SETCOLOR 2, _, _	Finestra testo.	POKE 710, _
SETCOLOR 1, _, _	Intensità caratteri della finestra testo.	POKE 709, _
SETCOLOR 0, _, _	COLORE 1.	POKE 708, _
SETCOLOR 1, _, _	COLORE 2 comanda anche l'intensità dei caratteri della finestra testo.	POKE 709, _
SETCOLOR 2, _, _	COLORE 3 nonchè colore della finestra testo.	POKE 710, _

Modi Grafici 4 e 6

SETCOLOR 4, _, _	Colore di fondo.	POKE 712, _
SETCOLOR 0, _, _	COLORE 1.	POKE 708, _
SETCOLOR 1, _, _	Intensità caratteri della finestra testo.	POKE 709, _
SETCOLOR 2, _, _	Finestra testo.	POKE 710, _

Modo Grafico 8

NOTA: Il colore della finestra testo è lo stesso del colore di fondo.

SETCOLOR 2, _, _	Finestra testo.	POKE 710, _
SETCOLOR 1, _, _	Solo comando dell'intensità del COLORE 1. Il colore è lo stesso del fondo.	POKE 709, _
SETCOLOR 4, _, _	Colore del bordo.	POKE 712, _

Modi Grafici 9, 10 e 11 (modi della GTIA); verranno trattati nel capitolo sulla GTIA.

Sempre più avanti con la GTIA

Nello sforzo continuo di migliorare le prestazioni dei suoi prodotti, la Divisione Home Computer dell'ATARI ha introdotto nei suoi computer un nuovo *chip*, chiamato GTIA.

Questo chip vi consente di scrivere in Basic programmi di grafica avanzata, fornendovi l'accesso a un maggior numero di colori e sfumature, in modi grafici aggiuntivi. In uno dei modi della GTIA avete ben 256 colori tra cui scegliere!

Modo grafico 9 (GTIA)

Vi consente di utilizzare simultaneamente 16 livelli di intensità luminosa per un singolo colore, sì che potete creare immagini sullo schermo con sottili differenze di tono.

Modo grafico 10 (GTIA)

Vi consente di mettere simultaneamente sullo schermo otto diversi colori, di intensità qualsiasi.

```
10 GRAPHICS 10
20 FOR K=705 TO 712:POKE K,INT(RND(0)*255):NEXT K
30 COLOR INT(RND(0)*7)+1:PLOT INT(RND(0)*79),INT(RND(0)*191):DRAWTO 39,95
40 GOTO 20
```

NOTA: L'utilizzo dei numeri casuali (RND) è spiegato dettagliatamente nel capitolo sulle Funzioni.

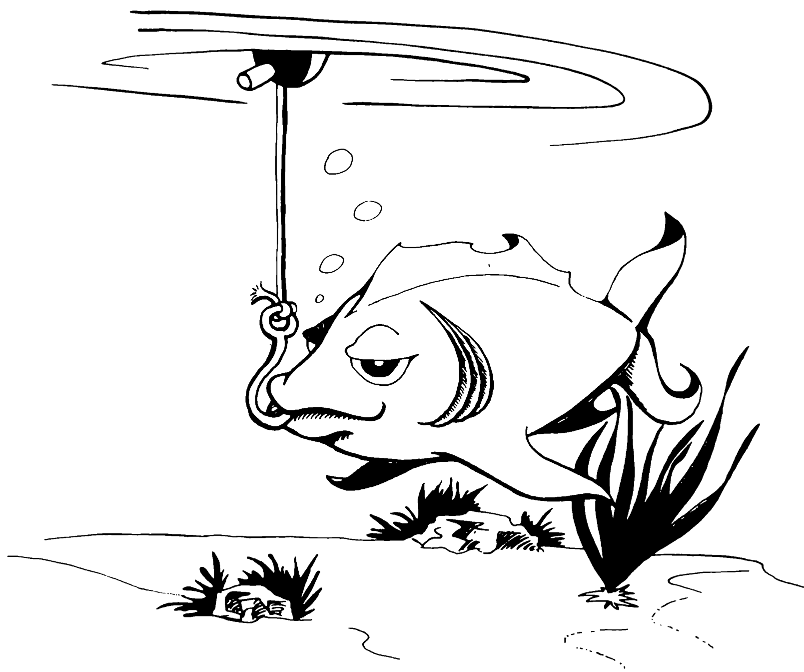
Modo grafico 11 (GTIA)

Vi fornisce la capacità di mettere sullo schermo simultaneamente 16 colori! La loro intensità luminosa può anche essere facilmente variata.

Per riassumere i tre modi addizionali della GTIA...

- **modo grafico 9:** scegliete uno dei 16 colori e potete utilizzarne 16 diverse sfumature.
- **modo grafico 10:** 8 colori, ma senza limitazioni su quale adoperare o sulla sua intensità. Colori chiari e scuri possono essere usati in tutte le combinazioni. Inoltre può essere pilotato anche il colore del fondo.
- **modo grafico 11:** 16 colori diversi sullo schermo, tutti nello stesso istante. Avete dinanzi uno spettro cromatico completo e scegliete uno o più dei sedici colori. Potete variare l'intensità con cui appare lo spettro, ma lo stesso livello di luminosità si applica a tutti i colori insieme.

Il chip GTIA dischiude al programmatore Basic principiante tutto il mondo del colore e della progettazione col computer; molte di queste possibilità erano sino ad ora note soltanto ai programmatori esperti ed alle persone che disponevano di costosi computer grafici.



"C'è una possibile trappola, ovviamente!"

Potreste non avere un chip GTIA nel vostro computer, ed avere al suo posto il chip originale CTIA.

Test di presenza della GTIA

Un modo facile per scoprire che tipo di integrato avete nel vostro computer consiste nell'eseguire questo semplice programma...

```
10 GRAPHICS 11
20 FOR Z=0 TO 15
30 COLOR Z
40 PLOT 0,Z:DRAWTO 79,Z
50 NEXT Z
60 GOTO 60
```

Se vi capita un qualsiasi messaggio d'ERRORE quando mandate in esecuzione il programma precedente, per prima cosa verificate di averlo copiato fedelmente. Se nella parte alta dello schermo compare un'immagine con gradevoli strisce di colore, stile arcobaleno, nel vostro computer è presente un chip GTIA. Se ciò non accade: o avete un televisore in bianco e nero, oppure avete un chip CTIA.

Se non avete la GTIA, ma ne volete assolutamente esplorare le possibilità grafiche, potete...

1. inviare il computer a un Centro di Servizio Autorizzato ATARI, per l'aggiornamento con aggiunta della GTIA

— oppure —

2. vendere il computer ad un amico cieco verso i colori e comprarvene uno nuovo.

Per mettere a confronto il modo grafico 11 (16 colori) e il modo grafico 9 (16 sfumature di uno stesso colore), potete semplicemente modificare la prima linea del programma precedente, in modo che si legga GR. 9 invece di GR. 11. Le differenze saltano all'occhio mandando in esecuzione il programma.

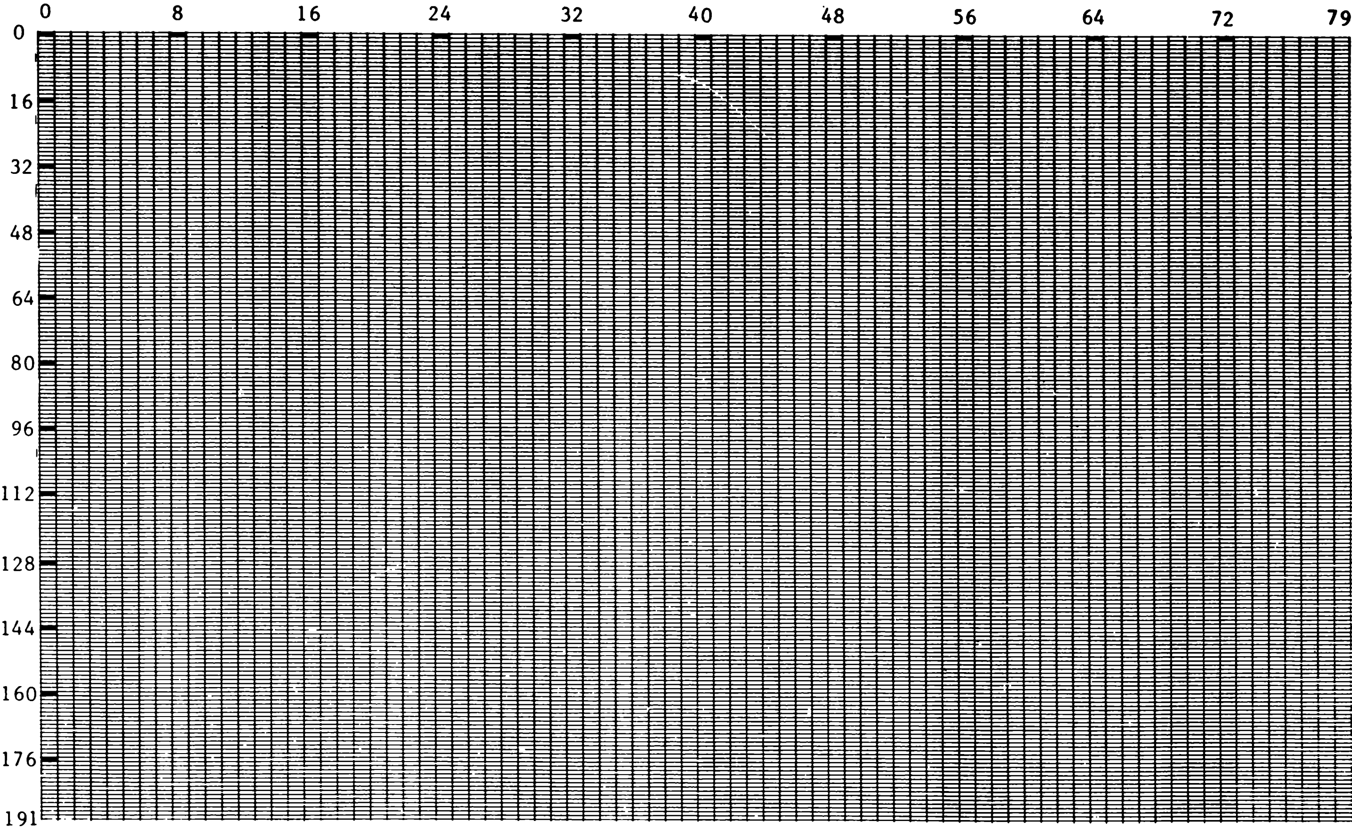
Come programmare in Basic i modi 9, 10, 11 della GTIA

Punto 1: Non esistono finestre testo nei modi GR. 9, 10 e 11 (e perciò niente *modo immediato*), così che dovete scrivere dei programmi se volete impiegare i modi della GTIA; questi sono modi totalmente grafici, e perciò non siete costretti ad aggiungere "+16" per eliminare la finestra testo.

Punto 2: La risoluzione dei modi grafici 9, 10 e 11 è la stessa: 80 × 192 pixel; risulta inusuale, in quanto ci sono più punti per il lungo che non per il largo.

Utilizzate il tracciato seguente per predisporre i vostri disegni che impiegano la GTIA.

Modo Grafico 9, 10 o 11



GTIA

Programmazione in GRAPHICS 9

Per selezionare il colore di base utilizzate la SETCOLOR come segue.

SETCOLOR 4, _, 0

Qui usate sempre 4.

Questo zero vi consente una gamma completa di sfumature.

Qui va un numero da 0 a 15; lo 0, per esempio, vi fornisce toni di grigio, mentre il 4 vi fornisce toni di rosso. Fate riferimento ai sedici colori base elencati nel capitolo sul colore.

Per selezionare l'intensità di un punto o di una linea del disegno, usate la COLOR seguita da un numero compreso tra 0 (minima luminosità) e 15 (massima luminosità).

NOTA: Se siete svegli dovreste aver osservato che il modo grafico 9 è un modo speciale che vi consente di scegliere tra 16 colori di 16 intensità, così che in totale avete da scegliere tra 256 colori!

Ecco un esempio che produce linee di due differenti toni di rosso su uno sfondo rosso scuro.

```
10 GRAPHICS 9
20 SETCOLOR 4,4,0
30 COLOR 12:REM LINEA ROSSO CHIARO
40 PLOT 39,2:DRAWTO 39,191
50 PLOT 40,2:DRAWTO 40,191
60 COLOR 4:REM LINEA ROSSO SCURO
70 PLOT 41,2:DRAWTO 41,191
80 PLOT 42,2:DRAWTO 42,191
90 GOTO 90
```

Il programma seguente disegna attraverso lo schermo una striscia sfumata con gradualità, passando per tutti i 16 colori, mediante la modifica del valore di mezzo (la variabile CO) della SETCOLOR di linea 100.

```
5 REM ESEMPIO IN GR.9
10 GRAPHICS 9
20 FOR Z=64 TO 80
30 COLOR Z
40 PLOT 2,Z:DRAWTO 79,Z
50 NEXT Z
60 FOR K=95 TO 80 STEP -1
70 READ X:COLOR X
80 PLOT 2,K:DRAWTO 79,K
90 NEXT K
100 FOR CO=0 TO 15:SETCOLOR 4,CO,0:FOR RIT=1 TO 300:NEXT RIT:NEXT CO:GOTO 100
110 DATA 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
```

Modo Grafico 10

Al pari dei modi grafici 9 ed 11, la risoluzione del modo grafico 10 è da 0 a 79 in orizzontale e da 0 a 191 in verticale. Il vantaggio di questo modo rispetto a tutti gli altri disponibili è che non solo potete scegliere 8 colori, oltre che la luminosità di ciascuno, ma anche un colore aggiuntivo per lo sfondo. Lo scotto che pagate a questa flessibilità è che il modo 10 è il più *infelice* per quanto riguarda la maniera di assegnare i valori al colore e alla luminosità. Per utilizzare il modo grafico 10 in Basic dovete ricorrere a COLOR, POKE e SETCOLOR.

La cosa più facile del procedimento sta nel fatto che utilizzate la COLOR da 0 a 8 per stabilire il colore dei punti colorati e delle linee da tracciare.

La parte difficile consiste nel mettere nei registri appropriati i colori specifici che si desiderano. La tabella seguente dovrebbe aiutarvi moltissimo:

MODO GRAFICO 10		VALORE DI DEFAULT PER LA POKE	LOCAZIONE PER LA POKE
COLOR 0	NO SETCOLOR	0	704 — sfondo
COLOR 1	NO SETCOLOR	0	705
COLOR 2	NO SETCOLOR	0	706
COLOR 3	NO SETCOLOR	0	707
COLOR 4	SETCOLOR 0,_,_	40	708
COLOR 5	SETCOLOR 1,_,_	202	709
COLOR 6	SETCOLOR 2,_,_	148	710
COLOR 7	SETCOLOR 3,_,_	70	711
COLOR 8	SETCOLOR 4,_,_	0	712

Il programma seguente disegna dapprima delle strisce di colore, quindi le modifica. Ciò mostra come possiate immettere sullo schermo 9 diversi colori simultaneamente.

```
10 GRAPHICS 10
20 FOR X=15 TO 65
30 N=INT(RND(0)*8)+1
40 COLOR N:REM ATTRIBUISCE ALLA LINEA UN COLORE A CASO (DA 1 A 8)
50 PLOT X,2:DRAWTO X,191
60 NEXT X
70 REM NEL SEGUITO VENGONO CONTINUAMENTE IMMESSI COLORI A CASO NELLE LOCAZIONI D
A 704 A 712
80 FOR PK=704 TO 712
90 K=INT(RND(0)*255)
100 POKE PK,K
110 NEXT PK
120 FOR RIT=1 TO 100:NEXT RIT
130 GOTO 80
```

Modo Grafico 11

Nel modo grafico 11 potete immettere sullo schermo 16 colori contemporaneamente; inoltre, potete pilotarne la luminosità.

In altri termini, nel modo GR. 11 è come disporre di un *arcobaleno* e di una manopola con cui si possa, a piacere, rendere l'arcobaleno più chiaro o più scuro.

Il colore della linea o del punto da disegnare viene definito con l'impiego della COLOR, da 0 a 15; i colori consistono nei 16 fondamentali, con una luminosità di partenza pari a 6.

```
10 GRAPHICS 11
20 FOR Z=0 TO 15
30 COLOR Z
40 PLOT 0,Z:DRAWTO 79,Z
50 NEXT Z
60 GOTO 60
```

Questo costituisce il vostro programma di base per il colore, e la modifica seguente vi fornirà qualche interessante risultato.

```
60 FOR LUMIN=0 TO 14:SETCOLOR 4,0,LUMIN:FOR RIT=1 TO 35:NEXT RIT:NEXT LUMIN:GOTO
 60
```

Per alterare l'intensità dei colori in GR. 11, si modifica la luminosità del registro 4 della SETCOLOR.

SETCOLOR 4,0, luminosità

Mettete qui un numero pari, da 0 (minima intensità) a 14.

Ecco ricapitolate le regole del Modo Grafico 11:

- Non disponete di finestre testo in fondo allo schermo.
- La risoluzione è di 80 (da 0 a 79) in orizzontale per 192 (da 0 a 191) in verticale.
- Potete immettere simultaneamente sullo schermo lo spettro dei 16 colori fondamentali, ma non siete obbligati ad usarli tutti se non vi serve.
- I colori sono selezionati specificando una COLOR da 0 a 15 prima di una PLOT o di una DRAWTO.
- Potete rendere più o meno scuri i colori modificando questo valore

SETCOLOR 4,0,_

ma non potete renderne alcuni chiari e altri scuri. Per far ciò è richiesto il modo GR. 10.

```
5 GRAPHICS 11
10 FOR Z=0 TO 79
20 PLOT 0,Z
30 COLOR Z
40 DRAWTO Z,79
50 NEXT Z
60 FOR Y=79 TO 0 STEP -1
70 PLOT Y,Z
80 COLOR Y
90 DRAWTO Y,79
100 NEXT Y
110 GOTO 10
```

```
10 GRAPHICS 11
20 LL=191
30 RL=79
40 FOR ZEBRA=0 TO 48
50 REM *CORSIA VELOCE* DELLA GTIA
60 COLOR ZEBRA
70 PLOT RL,LL
80 DRAWTO 0,0
90 RL=RL-1
100 NEXT ZEBRA
110 GOTO 110
```

```
5 GRAPHICS 11
10 FOR Z=1 TO 191
15 COLOR Z/12
20 PLOT 0,Z
40 DRAWTO 79,Z
50 NEXT Z
60 FOR FL=0 TO 14 STEP 2:SETCOLOR 4,0,FL:FOR RIT=1 TO 50:NEXT RIT:NEXT FL
70 FOR FLN=14 TO 0 STEP -2:SETCOLOR 4,0,FLN:FOR RIT=1 TO 50:NEXT RIT:NEXT FLN:GO
TO 60
```

Impiego dei Joystick e dei Controllori di Paddle

I joystick e i controllori di paddle (*manopole*) sono gli stessi impiegati nei Video Giochi ATARI. Se volete giocare col vostro computer, essi sono parte indispensabile del sistema: ma sono anche simpatici da provare ed assai utili in programmi educativi. Per aiutarvi ad utilizzare tali controlli, l'ATARI mette nel linguaggio Basic degli *aiuti alla programmazione* per joystick e paddle.

Vantaggi del joystick

Il joystick è molto utile quando volete movimenti di alto-basso e di sinistra - destra.

Nei suoi tempi morti, il computer ATARI verifica se per caso qualcuno non stia giocherellando con i controlli del joystick o delle paddle. E fa questo *60 volte* ogni secondo!

Il computer interpreta la posizione del joystick sotto forma di certi numeri. Ricordate, questi numeri sono predefiniti: non potete programmarli, ma dovete dire al computer che cosa fare se trova un particolare numero.

Provate questo esempio...

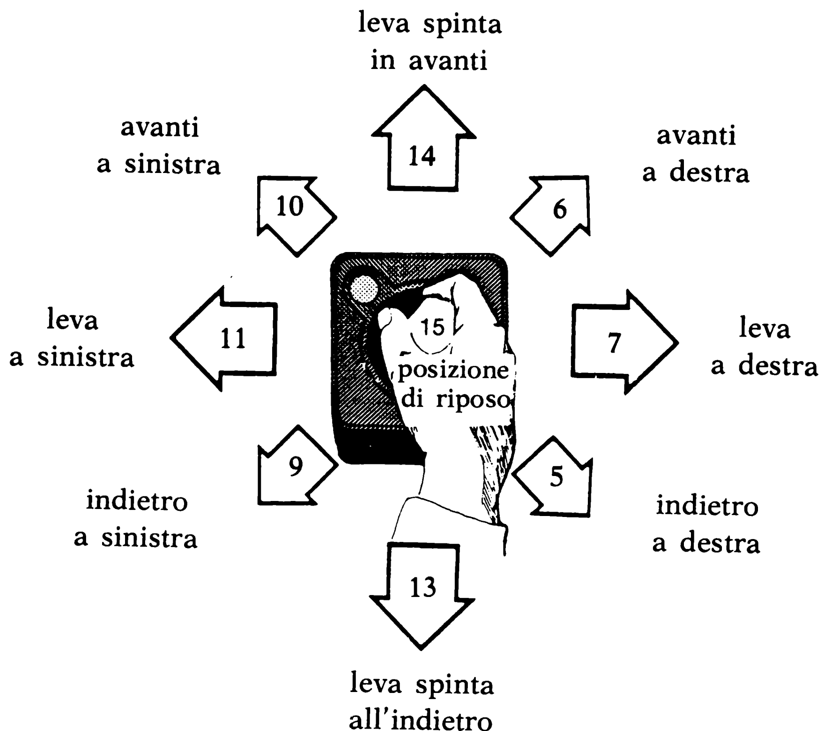
```
10 IF STICK(0)=14 THEN PRINT "AVANTI A TUTTA FORZA!"
20 GOTO 10
```



Disponete il joystick con il bottone di sparo collocato nell'angolo in alto a sinistra.

Adesso mandate in esecuzione il programma ed osservate cosa succede quando spingete la leva in avanti.

Il computer legge 14 quando avverte che il joystick è stato spinto in avanti; gli altri 8 numeri che esso utilizza sono i seguenti...



Provate questo programma:

```
10 PRINT STICK(0) : GOTO 10
```

Eseguite il programma e muovete intorno la leva; oppure cambiatelo così...

```
10 X=STICK(0)
20 PRINT X
30 SOUND 0,X*10,10,10
40 GOTO 10
```

NOTA: Talvolta è più facile lavorare con il joystick assegnandogli un nome di variabile ... X = STICK(0).

Ecco un programma che potete utilizzare per insegnare a qualcuno i punti cardinali.

```
10 PRINT "↖":POKE 752,1
15 FOR RIT=1 TO 100:NEXT RIT:X=STICK(0):IF X=15 THEN GOTO 120
20 IF X=14 THEN POSITION 17,2:PRINT "NORD"
30 IF X=13 THEN POSITION 17,22:PRINT "SUD"
40 IF X=11 THEN POSITION 2,12:PRINT "OVEST"
50 IF X=7 THEN POSITION 34,12:PRINT "EST"
60 IF X=6 THEN POSITION 29,4:PRINT "NORD EST"
70 IF X=5 THEN POSITION 29,20:PRINT "SUD EST"
80 IF X=10 THEN POSITION 2,4:PRINT "NORD OVEST"
90 IF X=9 THEN POSITION 2,20:PRINT "SUD OVEST"
110 GOTO 15
120 POSITION 2,12:PRINT "NON AVRAI UNA DIREZIONE NELLA VITA":PRINT "FINCHE´ NON
MUOVERAI IL JOYSTICK"
125 FOR RIT=1 TO 500:NEXT RIT
130 PRINT " ":GOTO 15
```

STRIG

STRIG è la contrazione di STICK TRIGger. Il computer verifica anche se premete il bottone rosso o *"trigger"* *(grilletto)* del joystick. Quando il bottone è premuto, il computer rileva il numero *zero* invece del numero 1.

Eseguite il programma seguente e premete il bottone del joystick per farvene un'idea.

```
10 PRINT STRIG(0)
20 GOTO 10
```

Oppure, per realizzare un sempice video gioco tipo tiro a segno, digitate e mandate in esecuzione questo programma:

```
10 REM SPARI
20 GRAPHICS 3+16
30 SETCOLOR 0,2,14:REM INTENSIFICA IL PUNTO
40 IF STRIG(0)=0 THEN GOTO 60
50 GOTO 10
60 FOR T=0 TO 38:SOUND 0,T,8,10
70 COLOR 1:PLOT T,12:COLOR 0:PLOT T,12:NEXT T
80 SOUND 0,0,0,0
90 GOTO 40
```

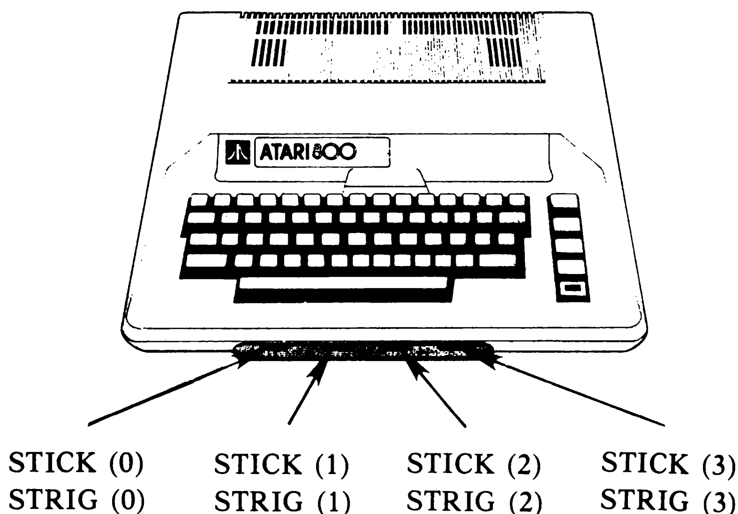
Un ultimo programma da provare è questo programma di disegno interattivo sul video:

```
10 GRAPHICS 7+16:X=80:Y=40
20 ST=STICK(0)
30 IF ST=7 THEN X=X+1
40 IF ST=6 THEN X=X+1:Y=Y-1
50 IF ST=14 THEN Y=Y-1
```

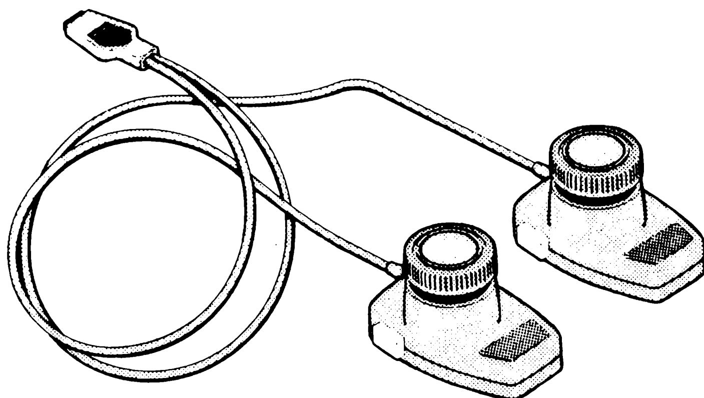
```
60 IF ST=5 THEN X=X+1:Y=Y+1
70 IF ST=11 THEN X=X-1
80 IF ST=10 THEN X=X-1:Y=Y-1
90 IF ST=13 THEN Y=Y+1
100 IF ST=9 THEN X=X-1:Y=Y+1
110 IF X>158 THEN X=159
120 IF Y>94 THEN Y=95
130 IF X<1 THEN X=0
140 IF Y<1 THEN Y=0
150 COLOR 1
160 PLOT X,Y
170 FOR RIT=1 TO 5:NEXT RIT
180 COLOR 0
190 IF STRIG(0)=0 THEN COLOR 1
200 PLOT X,Y
210 GOTO 20
```

NOTA: Disegnerete solo quando il bottone del joystick è premuto, e potrete anche cancellare le linee ripassandole senza premere il bottone.

Ricordate che potete scrivere programmi che utilizzano più di un joystick, a patto di individuarli propriamente.



Impiego dei Controllori di Paddle



Notizia molto importante: perché questi oggetti sono chiamati controllori di paddle *(lett.: controllori di racchette)?* Sono chiamati così semplicemente perchè assomigliano ai controllori utilizzati nel vecchio video gioco "PONG"; da cui: *controllori di "racchette (da ping-pong)".* Tuttavia scoprirete che hanno molti altri usi oltre quelli legati al gioco.

La programmazione delle paddle è simile a quella del joystick; con la differenza che, invece delle 9 posizioni del joystick, il computer riconosce la posizione della manopola tra 1 e 128. Per vederlo in azione, mandate in esecuzione questo programmino e *"smanettate"* sulla paddle.

```
10 PRINT PADDLE(0):GOTO 10
```

Come per il joystick, il bottone della paddle restituisce solo uno dei due numeri... 1=non premuto e 0=premuto.

```
10 PRINT PADDLE(0),PTRIG(0)
20 GOTO 10
```

Adesso, lasciateci fare un po' di musica:

```
10 X=PADDLE(0)
20 SOUND 0,X,10,10
30 GOTO 10
```

E che dire di un pochino di grafica e di colore...

```
10 SETCOLOR 2,A,2:COLOR 1
20 LET A=PADDLE(0)/6.2
30 POKE 82,A:PRINT "**"
40 SOUND 0,A,10,10
50 IF PTRIG(0)=0 THEN SOUND 0,A,6,A
60 GOTO 10
```

Le Funzioni

Le funzioni sono cose che vorreste il computer facesse automaticamente... e le fa!

Le calcolatrici a quattro funzioni eseguono addizione, sottrazione, moltiplicazione e divisione.

Calcolatrici più complesse hanno funzioni automatiche più evolute, come radici quadrate e percentuali. I piccoli computer, tuttavia, sono capaci a richiesta di creare senza sforzo praticamente qualsiasi *funzione* o *subroutine* (sottoprogramma). Nel *set* di istruzioni del Basic ATARI sono inserite soltanto alcune funzioni, tra le più complesse e le più frequentemente richieste.

Tutte le funzioni del Basic ATARI, comprese quelle trigonometriche, sono descritte nel Manuale di Riferimento del linguaggio: che dovreste consultare, prima di affrontare un qualsiasi programma sofisticato sul piano matematico.

Qui presentiamo solo un'introduzionone generale ad alcune delle funzioni che potreste usare in un primo apprendimento del Basic.

Alcune funzioni o *routine* automatiche che troverete utili o interessanti sono...

INT, RND, SQR, ASC, CHR$, e LEN

INT

INT sta per INTEGER *(intero)*. Questa funzione vi restituisce un numero intero al posto di un numero frazionario o decimale; arrotonda sempre il numero all'intero più vicino inferiore. Usate la INT come segue...

PRINT INT(5.2340)   vi restituisce ... 5
PRINT INT(9/2)   vi restituisce 4 invece di 4,5.

Potreste chiedervi che necessità ci sia di arrotondare i numeri, quando il computer non compie alcun lavoro aggiuntivo a tenerli nella loro forma esatta; eccovi qualche motivo: la casa media ha 23,212 finestre; la vostra automobile compie 43,12317 chilometri al giorno; e ovviamente, secondo un dato storico, l'Italiano medio ha 5,2 bambini.

Questo è il tipo di informazioni che la gente è abituata a ricevere dai computer e, benché accurate sul piano matematico, talvolta fanno apparire *un po' fessi* voi e il vostro computer; INT vi aiuta a mantenere semplici i risultati semplici.

Inoltre, con INT potete verificare se un numero è pari o dispari. Esaminate il frammento...

```
10 INPUT X
20 IF X/2=INT(X/2) THEN PRINT "IL NUMERO E´ PARI"
```

Sia per esempio X=9; in tal caso X/2 sarebbe uguale a 4,5 e INT(X/2) risulterebbe 4 (dall'arrotondamento di 4,5); perciò il computer non potrebbe davvero scrivere: "IL TUO NUMERO È PARI".

Ma non dovete credere a me, chiedetelo al computer.

```
10 PRINT "↖":PRINT "BATTI UN NUMERO SEGUITO DA RETURN"
20 INPUT X
30 IF X/2=INT(X/2) THEN PRINT "IL NUMERO E´ PARI":FOR RIT=1 TO 400:NEXT RIT:RUN
40 PRINT "E´ DISPARI... RIPROVA!":FOR RIT=1 TO 400:NEXT RIT:RUN
```

Troverete altri impieghi per la funzione INT, tra cui l'utilizzo congiunto con la funzione RND.

RND

RND sta per "RANDOM NUMBER" *(numero casuale)*. Un numero casuale ha la stessa probabilità di verificarsi, o di essere estratto, di qualsiasi altro numero. Dopo che è stato scelto, il numero viene reinserito nel mucchio; potrebbe essere ripescato immediatamente o dopo un tempo lunghissimo. È un fatto del tutto legato al caso.

Ci sono due modi per procurarsi numeri casuali col computer ATARI: un modo semplice è eseguire la PEEK di una locazione che produce dati casuali tra 0 e 255; l'altro è utilizzare la funzione RND che li fornisce appositamente.

Cominciamo con la PEEK.

```
10 X=PEEK(53770)
20 PRINT X:GOTO 10
```

Mandate in esecuzione il programma per vedere i risultati. Adesso, *musica maestro...*, aggiungete

```
15 SOUND 0,X,10,10
```

I numeri casuali che voi recuperate con la PEEK sono facili da usare, ma non sono altrettanto flessibili di quelli procurati dalla funzione RND.

X=INT(RND(0)*10)+1

INT	0	10	+1
INT arrotonda il risultato	Questo non cambia mai	Qui mettete il numero casuale massimo che volete	Questo serve a sbarazzarsi dello zero e rendere possibile il 10, poichè la INT arrotonderebbe a 9 un numero come 9,89984372.

Provate con...

```
10 X=INT(RND(0)*10)+1
20 PRINT X
30 GOTO 10
```

Adesso modificate la linea 10 in...

10 X=INT(RND(0)*1000)+1

Nel semplice gioco che segue, il computer estrae un numero tra 1 e 25 e voi tentate di indovinarlo.

```
10 PRINT "↖":POKE 752,1
20 X=INT(RND(0)*25)+1
30 PRINT "STO PENSANDO UN NUMERO TRA 1 E 25...  FAI UN TENTATIVO E PREMI RETURN"
40 POSITION 18,11:INPUT G
50 IF X=G THEN GOSUB 80:POSITION 16,7:PRINT "GIUSTO!":FOR Z=0 TO 254 STEP 2:POKE
 712,Z:SOUND 0,Z,10,10:NEXT Z:RUN
60 IF X>G THEN POSITION 8,17:PRINT "TROPPO BASSO... RIPROVA!":POSITION 18,11:PRI
NT "    ":GOTO 40
70 IF X<G THEN POSITION 8,17:PRINT "TROPPO ALTO... RIPROVA!":POSITION 18,11:PRIN
T "    ":GOTO 40
80 POSITION 8,17:PRINT "                              ":RETURN
```



SQR

La funzione SQR vi fornisce la RADICE QUADRATA *(SQuare Root)* di un numero. È impiegata in questo modo...

PRINT SQR(100) vi darà 10
PRINT SQR(25) vi darà 5

Ricordatevi, SQR non serve per fare il quadrato di un numero; per far ciò utilizzate invece:

PRINT 10ˆ2 vi dà 99.99999598

oppure

PRINT 5ˆ2 vi dà 24.9999993

NOTA: Ogni computer utilizza formule diverse per eseguire i quadrati. Se vi serve un quadrato *esatto*, potete aggiungere 0,5 al risultato così da portarlo oltre l'intero immediatamente superiore e quindi arrotondarlo con la funzione INT.

```
10 PRINT "↰":POKE 752,1:PRINT "NUMERO DI CUI VUOI IL QUADRATO"
20 INPUT X:POSITION 7,11:PRINT "IL QUADRATO DI ";X;" E'... ";INT((X^2)+0.5)
30 FOR RIT=1 TO 700:NEXT RIT:RUN
```

ASC e CHR$

Ad ogni lettera, numero o carattere della tastiera del vostro Computer ATARI è associato un codice preciso. Per i numeri ed i tasti ordinari il codice è lo standard industriale ASCII *(si pronuncia "aski")*.

Per i caratteri speciali si utilizza un versione estesa del codice ASCII, detta ATASCII dalla contrazione di ATARI-ASCII.

Per vedere il codice associato alla lettera A, digitate...

PRINT ASC("A")

Quando premete RETURN dovreste ottenere 65. E se adesso digitate...

PRINT CHR$(65)

avrete in risposta la lettera A.

(Questa istruzione dice al computer di stampare il carattere di codice ASCII n. 65).

NOTA per la STAMPANTE: poiché molto codici sono specifici per gli Home Computer ATARI, sulle stampanti di molti computer non è possibile stampare l'intera lista dei CHR$.

Viceversa, sullo schermo

(qui va una qualsiasi lettera o carattere)

PRINT ASC(" ") vi darà il codice numerico

e

(qui va un numero qualsiasi da 0 a 255)

PRINT CHR$( ) vi darà il carattere corrispondente al codice numerico

Usate il programmino seguente se volete vedere (ed udire) tutti i caratteri disponibili.

```
10 FOR Z=0 TO 255
20 PRINT CHR$(Z)
30 NEXT Z
```

Potrete osservare che uno dei codici fa suonare il cicalino del computer...precisamente il CHR$(253).

Provate questo programma da mal di testa...

```
10 PRINT CHR$(253):GOTO 10
```

Per battere sulla tastiera, ricavandone prima il codice del tasto e quindi la rappresentazione corrispondente CHR$, potete usare il seguente programma:

```
10 GRAPHICS 2+16:PRINT #6;"MAESTRO DATTILOGRAFO"
20 OPEN #1,4,0,"K:"
30 GET #1,X
40 POSITION 9,5
50 PRINT #6;CHR$(X)
60 SOUND 0,30,10,4:FOR RIT=1 TO 75:NEXT RIT:POSITION 9,5:PRINT #6;" ":SOUND 0,0,
0,0
70 GOTO 30
```

NOTA: PRINT CHR$(34) produrrà sullo schermo una *coppia di virgolette*

mentre

i codici ASCII possono essere usati per scandire le lettere delle parole: la A maiuscola è 65, la B è 66, ecc.

LEN

LEN vi dice di quanti caratteri è composta una variabile stringa: PRINT LEN(A$).

```
10 DIM NOME$(20):PRINT "DIMMI IL TUO NOME":INPUT NOME$
20 PRINT "CI SONO ";LEN(NOME$);" LETTERE NEL TUO NOME"
```

oppure potete usare LEN per centrare la scritta...

```
5 PRINT "↖":POKE 752,1
10 DIM NOME$(30):PRINT "NOME E COGNOME":INPUT NOME$
20 PRINT "↖":POSITION (40-LEN(NOME$))/2,11:PRINT NOME$
30 PRINT :POSITION 15,13:PRINT "E' CENTRATO"
40 GOTO 40
```

NOTA: per ulteriori informazioni sulle funzioni dei Computer ATARI, consultate il Manuale di Riferimento del Basic ATARI.

Messaggi d'ERRORE

Quella che segue è una ristretta panoramica dei più comuni messaggi d'ERRORE. Per dettagli più estesi, consultate il Manuale di Riferimento del Basic ATARI e del DOS II ATARI.

Messaggi d'ERRORE comuni

N. errore	Probabile causa	Alcuni rimedi
2	*memoria RAM insufficiente*	– procuratevi più RAM – imparate a programmare in maniera più efficiente – imparate l'assembler



N. errore	Probabile causa	Alcuni rimedi
3	*errore su un dato:* avete tentato di fare qualcosa del tipo mettere un numero negativo in un'istruzione del suono o dividere per zero.	– non fatelo.
4	*troppe variabili,* il massimo è 128.	– riducete le variabili – usate gli ARRAY.
6	*errore "out of data":* non avete un flag adatto in un'istruzione DATA; oppure, se si è verificato un errore 6 senza numero di linea, avete premuto il tasto RETURN mentre il cursore era posto sopra la scritta READY dello schermo.	– nel primo caso mettete un flag ed utilizzatelo propriamente. – nel secondo caso, imparate a non preoccuparvi: non succede niente.
7	*numero di linea* del programma *oltre 32767*	– macchina indietro!
8	*INPUT non corretto:* state introducendo variabili numeriche o variabili stringa?	– controllate e correggete.
9	*errore su stringhe o ARRAY:* una variabile non è mai stata DIMensionata, o è DIMenesionata due volte; oppure ha una DIMensione eccessiva, p. es. DIM A(5780)	– Dimensionate stringhe, array e matrici in cima al programma e non ritornateci mai con dei GOTO.
12	*linea màncante:* avete forse modificato alcuni numeri di linea, dimenticando di cambiare di conseguenza GOTO e GOSUB.	– evitate di far riferimento a una linea inesistente.

N. errore	Probabile causa	Alcuni rimedi
13	*incontrato un NEXT senza il FOR* corrispondente.	– controllate il programma.
16	*errore su Return.*	– potreste aver cancellato un GOSUB dimenticando il RETURN.
130	il *dispositivo* indicato è *inesistente* o non in linea.	– controllate che le periferiche siano propriamente connesse.
138	*DEVICE TIME OUT:* può succedere per molti motivi e qualche volta è fastidioso	– collegate eventuali cavi sconnessi. Accendete tutti i dispositivi spenti. Con programmi su cassetta, riavvolgete la cassetta e riprovate.
139	*NAK ("not acknowledged": non riconosciuto),* riferito a un certo dispositivo periferico.	– riprovate, come nel caso dell'errore 138.
141	*cursore fuori limiti:* avete posizionato il cursore fuori schermo o oltre l'area indirizzabile nel modo grafico corrente (p. es. in GR. 3 non potete eseguire una PLOT 45,20 e in GR. 2 non potete fare POS. 2,15).	– questo può richiedervi del tempo; mettetevi a cercare e probabilmente scoprirete dove siete usciti dai limiti.
143	*Checksum ERROR:* errore sulla verifica di correttezza del dato (qualcosa non va)	– Con programmi su cassetta, riavvolgete la cassetta e riprovate.

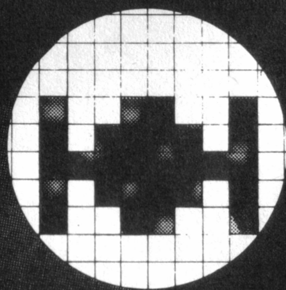
N. errore	Probabile causa	Alcuni rimedi
144	*problemi in lettura o scrittura* di dischetti	– verificate se non avete protetto il disco contro la scrittura (tacca coperta). Tentate di rieseguire il programma; il vostro drive potrebbe essere lievemente incompatibile col dischetto.
147	*Memoria RAM insufficiente per il modo grafico.*	– non usate quel modo grafico – confrontate l'errore n. 2.
170	*File non trovato:* l'avete scritto bene? Oppure... avete un punto (.) tra il nome del file (del programma) e la sua estensione? O è danneggiato il dischetto?	– se avete un disco, digitate DOS RETURN A RETURN RETURN (verificate la correttezza del nome del file); digitate B RETURN – Tentate di rieseguire il programma

Conclusione

Mi sono sforzato di includere in questo libro soltanto cose necessarie, gradevoli o interessanti. Per la natura stessa dell'argomento, temo però di aver tralasciato un po' delle prime e molte delle seconde. C'è la speranza di avervi stuzzicato l'appetito, dandovi un piccolo aiuto per comprendere alcune delle possibilità dell'Home Computer ATARI.

Esistono molti bei libri ed articoli già pubblicati su specifiche caratteristiche dei Computer ATARI quali gli *interrupt* della *display list*, lo *scrolling* fine e la grafica *player-missile:* non sono argomenti facili da dominare; ma se siete stati pazienti a sufficieneza da *digerire* con successo quanto qui presentato, lanciatevi pure!





INDICE ANALITICO

Un ottimo libro per tre categorie di lettori. Prima di tutto coloro che non hanno alcuna esperienza di programmazione nè di computer, e d'altra parte non desiderano passare i prossimi dieci anni a farsela.
Poi coloro che sanno tutto sul BASIC ma non sul BASIC ATARI; come certamente sanno ogni calcolatore è diverso, e questo libro consentirà loro di conoscere ben presto le caratteristiche ATARI, saltando le parti comuni a tutti i BASIC.
Infine i collaudati programmatori ATARI, che sanno tutto sul BASIC, e anche sul BASIC ATARI, e potranno usare il testo come manuale di riferimento, sintetico e veloce da consultare.
Il libro è semplice, chiaro e preciso; evita dettagli tecnici non necessari e la terminologia oscura di molti libri analoghi; insomma è il compagno ideale per il vostro calcolatore ATARI.

L. 18.000 Cod. 540H 88-7056-173-9



GRUPPO
EDITORIALE
JACKSON

Bill Carris

BASIC ATARI

156